AN IMAGE-BASED APPROACH FOR 3D RECONSTRUCTION OF URBAN SCENES USING ARCHITECTURAL SYMMETRIES

A Dissertation
Presented to
The Academic Faculty

By

Natesh Srinivasan

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of College of Computing

Georgia Institute of Technology

August 2018

AN IMAGE-BASED APPROACH FOR 3D RECONSTRUCTION OF URBAN SCENES USING ARCHITECTURAL SYMMETRIES

Approved by:

Dr. Frank Dellaert
Interative Computing
Georgia Institute of Technology

Dr. Irfan Essa
Interative Computing
Georgia Institute of Technology

Dr. James Hays
Interative Computing
Georgia Institute of Technology

Dr. James Rehg
Interative Computing
Georgia Institute of Technology

Date Approved: July 3, 2018

Dr. Yanxi Liu
Computer Science and Engineering
Pennsylvania State University

Symmetry is what we see at a glance; based on the fact that there is no reason for any difference...

–Blaise Pascal

– to Amma and Appa

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my advisor whose infinite patience with me has not only helped me with my thesis but also profoundly changed the way I learn and understand problems. Secondly, my parents who have firmly stood behind my decision to spend the better part of my life being a student. And last but not the least, my friends who take away my worries with their never ceasing cheer. A special acknowledgement to Karthik Veluswamy for helping me with the dataset collection of the Royal Albert Hall in London. I would also like to thank my labmates especially Luca Carlone for his involvement in the earlier part of my work and his many useful insights into optimization problems. I would also like to thank Richard Roberts for the hours of discussions on a variety of problems that has helped me become a better student and researcher.

TABLE OF CONTENTS

LIST OF FIGURES

SUMMARY

In this dissertation, I focus on an important, generalizable and freely available sub-category of semantic information in addressing modern reconstruction challenges: the notion of symmetry. The emphasis in the 3D modeling of urban scenes has shifted in the past decade. The current goal of the reconstruction community is to provide dense, CAD-like representations of the 3D built environment. In my thesis I focus on five main contributions exploiting symmetry in 3D scenes that advance this agenda:

I provide a framework for modeling complex symmetries in 1D, 2D and 3D built upon the mathematical theory of symmetry and lattices, with an emphasis on how this applies to urban scenes. Drawing largely on crystallographic theory, but also considering lattice-free symmetries, my intent was to create a firm basis for computer vision applications in my thesis and beyond.

I develop a probabilistic modeling framework based on Bayes' networks that provides a set of generative models for symmetric scenes. Doing so allows us to exploit the physical meaning of the variables that are generating the symmetries to form probabilistic priors. In addition, it serves as the foundational basis of a factor graph framework for optimizing the parameters of symmetry and its subsequent use in 3D reconstruction methods.

I provide a novel voting scheme in a polar transformation space to determine the lattice parameters of a symmetric scene that have a lattice-like structure. I demonstrate that my algorithm is more robust to the variations in the quality of the point clouds generated from Structure from Motion (SfM) algorithms, as compared to state of the art techniques that vote in a Cartesian coordinate space.

By exploiting the full generative modeling introduced earlier, I show that one can obtain a detailed 3D reconstruction from a single image of a building or structure, while simultaneously determining the camera parameters. I discuss the interaction of the generative model of symmetry in an image projection space and show how one can infer the complex 3D symmetry of the scene using 2D measurements in only a single image, extending previous work in this area to more general symmetries.

Finally, I address the topic of joint SfM and symmetry detection. SfM is currently an indispensable tool in computer vision, but it is still not sufficiently agnostic to the type of environment that we are trying to reconstruct. Highly symmetric scenes present one such problem to model SfM methods. I show that instead of being a hindrance, symmetry can be a powerful constraint in providing dense and photo-realistic 3D models of the scene. In particular, I discuss three cases in which I use all the other contributions in this thesis to advancing the state of the art in 3D reconstruction to much more general symmetries than those that are hitherto considered.

Chapter 1

INTRODUCTION

The present goal of the 3D reconstruction community is to provide CAD-like models of the world that will immediately benefit emerging technologies like autonomous driving, augmented and virtual reality, software-aided architecture etc. The standards and nuances of 3D modeling of urban scenes has shifted in the past decade. While a majority of algorithms discussed in multi-view geometry literature [HZ00] are useful for obtaining a rudimentary reconstruction of the scene represented using geometric primitives such as 3D points [Aga+11], lines [SD06] or planes [SSS09], these reconstructions are not directly useful for the tasks that I have listed above such as navigation, civil engineering and architecture. Therefore, recent works have focused on transforming reconstruction techniques into industry-quality, CAD-like outputs that are denser [NLD11], photo-realistic [Sha+14], and semantically accurate with the help of complimentary information such as, image segmentation [BS11], learned priors [Bao+13], assumptions of smoothness/planarity [Sal+14], or databases of 3D models [SX14] etc.

In this work, I focus on an important, generalizable and freely available sub-category of semantic information - the notion of *symmetry* and its utility in addressing the above modern

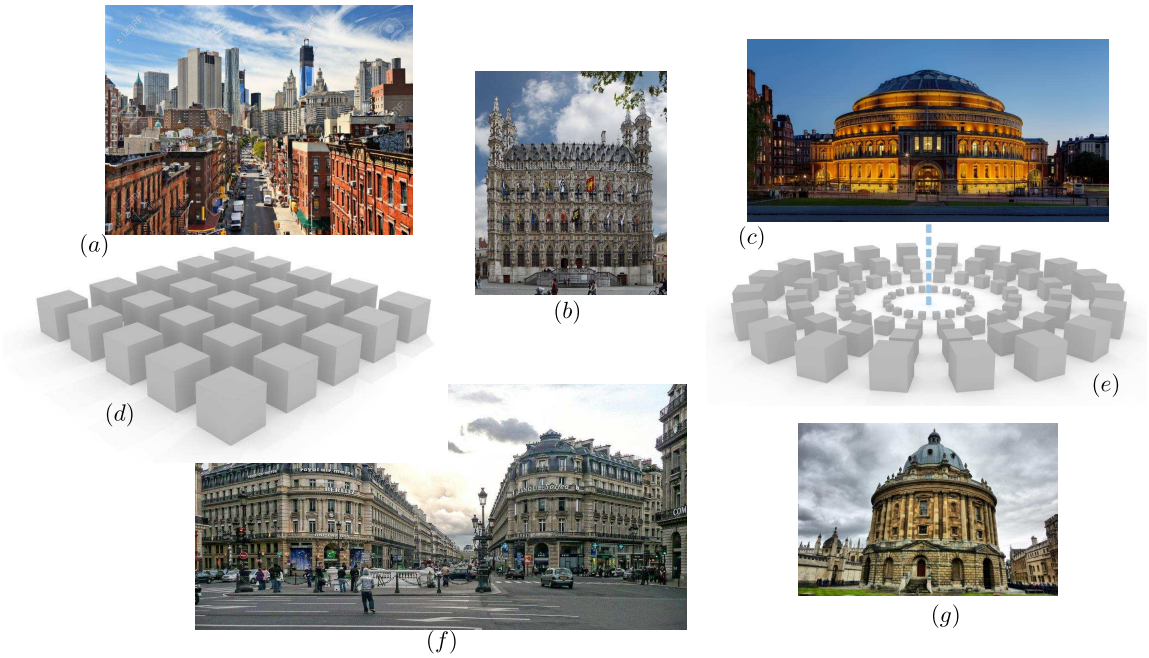


Figure 1.1: **Problem Definition**: **Reconstruction of Urban scenes that exhibit symmetries**. (a) Examples of urban scenes exhibiting translational symmetries (from Lower Manhattan, New York) (b) Leuven and (f) Paris. (c) On the right we have images of the Royal Albert Hall and (g) the Radcliffe camera building from Oxford, exhibiting rotational symmetries. Our goal is to identify 3D symmetries and obtain a reconstruction of the scene whose individual geometric primitives(3D points) behave according to the rules of symmetries that are identified, for example (d, translational) and (e, rotational).

reconstruction challenges. How humans exploit symmetry in understanding the environment is not fully understood, but, it is clear that it plays an important role in our cognitive process [CN89]. The role of symmetry in multi-view reconstruction has been studied before by several other researchers [Hon+04; Fun+17; LL07]. However, much of these related works are not general enough and the focus of these approaches are often restricted to specific symmetries like translational symmetries [Cey+14; JTC09; Coh+12] with very few works for detection of rotational and reflective symmetries [Coh+12], and other com-

plex symmetries like rotoreflection [LL13], glide reflection [WCL16] and helical symmetries. Even in single view reconstruction techniques, there are no methods out there to ubiquitously discover symmetries and reason about the underlying geometric model, something human beings and most primates [Sas+05] are able to do well without having any prior knowledge about the scene. Our goal is to identify multiple, complex 3D symmetries in urban scenes with/without the aid of manual intervention, providing compact, denser and interactive 3D models that can benefit emerging technologies.

In this thesis, I address the problem of **image-based reconstruction** and **a symmetry-based description** of urban scenes. While the computer graphics community provides numerous parametric representations for shape, in this work, I will focus on a representation that captures the set of **3D Euclidean symmetries** exhibited by the scene. By using the symmetry in these scenes to my advantage, I provide **denser** and more **compressed** point clouds, rich in semantic information. Although my work deals exclusively with 3D models represented as points, my approach can be extended to other popular geometric primitives such as lines and planes since the definition of symmetry is agnostic to these choices provided the definition of isometries [LCT04] is valid for these as we will demonstrate in Chapter 2. In addition to this, I provide the notion of **asymmetric units,** the fundamental building blocks of symmetry, that can be used in conjunction with the parametric rules of symmetry to provide a generative description of the scene that can be immediately used for the tasks listed above, such as, architectural manipulations, augmented reality and other real-world applications.

Jointly addressing the task of detecting 3D symmetries and reconstruction could prove mutually beneficial for both these individual problems. Existing works that discover 3D

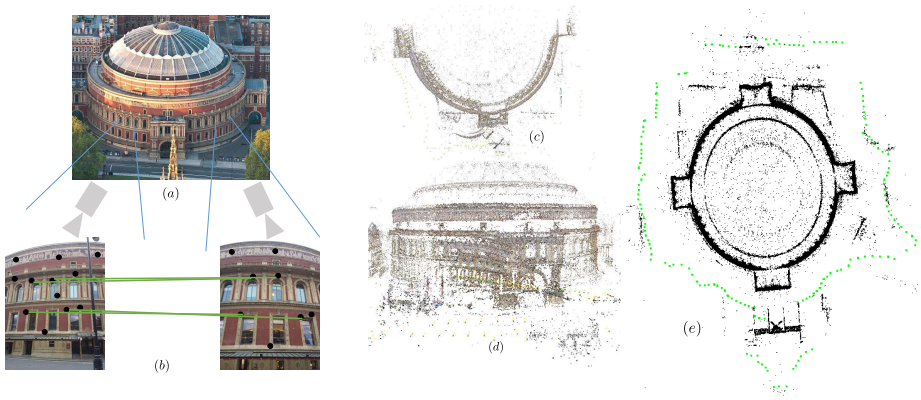


Figure 1.2: **Vanilla implementations of SfM fails under the presence of symmetry**: Reconstruction of the Royal Albert Hall in London using a series of images taken from a close distance using Bundler [Sna+10]; the building exhibits circular symmetry. (a) shows the top view of the building. (b) shows two images takes from a close distance with no overlap, yet feature matching between the two images yields correspondences that are clearly incorrect. Incorrect correspondences arise due to symmetry because local image features used for matching are not aware of the global symmetry exhibited by the scene. (d) The reconstructed 3D scene is grossly incorrect. Despite a full 360 degree path around the building, a lot of images are discarded away due to symmetry and the model also exhibits distortion in the fronto-parallel section. In (e) we show the top view of a semi-automatic, manually corrected sparse model of the scene that disregards symmetric matches and uses non-symmetric features to obtain a globally consistent camera pose(green) and structure(black).

symmetries using images are limited to simple symmetry groups with almost no works for detecting and describing complex 3D symmetries.

While symmetry provides a valuable source of information for 3D reconstruction, it can also cause problems for SfM because of incorrect correspondences between features, if treated naively. In addition, matches that are deemed "incorrect" in traditional SfM pipelines become relevant and carry important information for the joint task of symmetry detection and reconstruction. Figure 1.2(b) shows an example where two SIFT features are incorrectly matched between two windows in the traditional feature matching stage of

the SfM pipeline. Incorrect correspondences either discard away entire images during the RANSAC stage of SfM resulting in sparser, incomplete models where the obtained point clouds are far too sparse for detecting 3D symmetries, or, more significantly, if they are considered as valid matches, severely distort the obtained 3D models shown in Figure 1.2.

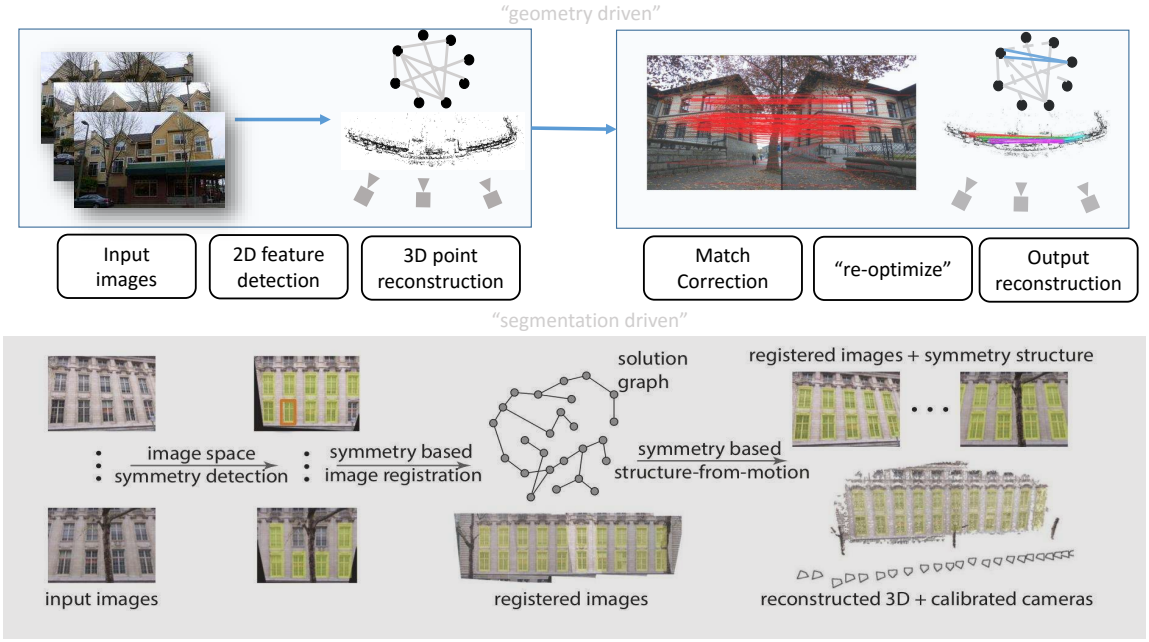


Figure 1.3: **Categorization of Existing Approaches** (Figure from [Coh+12]&[Cey+14]): We can categorize existing approaches that address the problem of reconstruction in the presence of architectural symmetries into (1) geometry driven (2) segmentation driven. [Top row (white)] geometry driven approaches [Coh+12; Rob+11]: The idea is to improve feature matching between image pairs using the scene geometry which require an initial reconstruction of the scene. [Bottom Row (gray)] segmentation driven approaches [Cey14]: manual intervention to segment out the repeating regions which are then identified across all the images. These identified regions are used to fix the initial geometry of the scene which are later used to establish reliable correspondences that can be used for symmetry detection.

Related works that address the joint task of reconstruction and detection of complex 3D

symmetries are limited in their scope. Jointly detecting complex 3D symmetries and performing 3D reconstruction using images is a hard combinatorial search problem in a discrete-continuous space and at the time of writing this thesis, I am only aware of one recent work that directly addresses this problem [Cey+14]. In this approach, Ceylan *et al.* [Cey+14] reduce the search space using a combination of manual intervention and higher level image segmentation technique to achieve a dense piecewise-planar model. Wolff *et al.* [WCL16]use successfully exploit lattice to aid in 3D reconstruction. Related works in the graphics community reason about the 3D symmetries by directly operating on dense 3D models [Mit+12], but this body of research cannot be applied directly to noisy (in many cases, distorted) point clouds obtained from SfM. Another closely related work is the area of near regular textures (NRT)[LLH04] which considers the pattern observed in the image as a texture that is obtained as a result of geometry, lighting and material properties and attempts to solve the underlying geometric model as a function of linear independent basis vectors called *generators*. However this method works well only if the entire image comprises of this pattern. Even if we segment out the region that contained these regular textures, there is no way to obtain the 3D relations between multiple textures that capture the underlying 3D geometry. NRTs have had some success in geo-tagging [Sch+08] which in-turn provide valuable structural constraints to the problem of SfM. Related multi-view geometry techniques focus on the problem of incorrect correspondence due to symmetry by using features that are invariant to specific symmetries [WFP10] or by modeling the notion of repetition into the optimization process [Rob+11; Coh+12]. Related works in the area of facade parsing papers include Cohen *et al.* [Coh+17; AD04; BGM07; BH09; Cey+12; BR06; CT99; Dai+12; DTM96], which do exploit translational symmetry in

many cases. However these methods focus on rectifying the sparse 3D model rather than identifying complex 3D symmetries and the elements that make it up. In this work we will address the drawback in these state-of-the art algorithms.

The **objectives** of this thesis are

1. To provide a classification of symmetry that is more useful for the computer vision community to identify unsolved problems while effectively being able to zero-in on relevant existing works to establish a baseline for comparison(similar to challenges such as [Fun+17]).

2. To develop a method that can determine generators of complex 3D symmetries directly from image(s) that are not restricted to translational symmetries alone [Cey+14] but**,** a more general class of 3D symmetries.

3. To provide a technique to determine the fundamental building blocks of symmetry, i.e the asymmetric units, which, in combination with the identified symmetry generators will allow us to define a generative model of the scene. Although 2D wallpaper symmetry detection techniques talk about "Motif Extraction" [LCT04](which are the 2D equivalent of asymmetric units), there is a lack of an equivalent technique in 3D and one of the primary motivation of this work is to bridge this gap in understanding between 2D and 3D symmetries.

4. To develop techniques that can circumvent the data association problem of SfM in the presence of symmetry by rigorously modeling symmetry into the data association process.

In this thesis, I develop multiple methods that solves the objectives enumerated in the previous paragraph. This thesis is divided into four parts - In the first part, I summarize the many great works in the theory of symmetry extracting those parts that are relevant to the topic of 3D reconstruction. In the second part, I provide a probabilistic generative model for synthesizing symmetries. In the third part, I address the problem of optimizing for lattices and reconstruction simultaneously. In the fourth part, I provide a method for determining the structure of a symmetric scene from a single image while extracting the asymmetric units. In the fifth and final part I provide a technique for extending the reconstruction of single views to multiple views.

I first summarize the relevant group theoretic aspects of symmetry obtained from authoritative texts to setup the platform for addressing the problem under consideration here, i.e, 3D reconstruction of symmetric structure. In particular, I make the following 4 claims in this thesis:

1. I provide generative models for synthesizing 3D points on symmetric structures in 1D, 2D and 3D.

2. I developed a method for determining the lattice parameters of those generative models from sparse and noisy SfM clouds by exploiting a novel polar space voting scheme.

3. Given a single view and a known symmetry type, I provide a method for determining the 3D structure of the scene as comprised of non-decomposable elements known as asymmetric units.

4. For the case of multiple views, I jointly perform structure from motion and determine the generative model of symmetry and simultaneously obtaining a dense 3D model of asymmetric units (**Motifs**).

My **contributions** in this thesis are as follows:

1. A generative model for complex symmetries that supersedes the insights provided by Liu *et al.* [LCT04] by addressing the larger space of 3D symmetries.
2. A novel polar space voting scheme, that can detect lattices from noisy reconstructed point clouds.
3. A novel technique for determining the 3D geometry of a symmetric scene from a single image while providing a descriptive model of the scene in terms of units called asymmetric units.
4. A novel camera calibration scheme that treats any symmetric scene as a calibration pattern.
5. A symmetry-aided Structure from motion algorithm that goes beyond state of the art works dealing only with lattices [Cey+14].
6. A novel dataset that challenges the computer vision community to find algorithms that can detect more complex 3D symmetries.

Chapter 2

CLASSIFICATION OF SYMMETRY

The contents of this chapter substantiate the first claim of my thesis:

> *I use the mathematical theory of symmetry from authoritative texts to synthesize an overview of symmetry that is immediately beneficial to the computer vision community.*

The theory of symmetry predates, in many ways, the theory of geometry itself due to its intuitive appeal to the human psyche. If Euclid laid the foundations of modern geometry, then, it was undoubtedly preceded by the discussions on symmetry by Plato. Symmetry has since been exploited as a valuable tool almost as indispensable as geometry itself in almost all scientific and engineering endeavors such as architecture, physics, biology, chemistry to name a few. Thanks to a variety of discoveries in these fields, a modern theory of symmetry has since grown to such an extent that it deserves a mathematical treatment of its own outside the context of its applications leading to discovery of its group theoretic properties that is not immediately apparent from an application standpoint. We enlist the opinions of authors such as [CS13; Wey15; Mar12; McW02; Mül13] who have deeply

thought about the topic of symmetry and we leverage their understanding to synthesize a view of symmetry that is beneficial to the computer vision community. While the collision of symmetry with computer vision has been relatively new in comparison, it is by no means small, specific instance of symmetries have been explored by many authors within the many micro-disciplines that span our field such as detection, recognition, reconstruction etc. There are several detailed literature on symmetry detection in the context of computer vision and its sister field, computer graphics. One of the more recent surveys in this topic was done by Liu *et al.* [Liu+10]. A literature survey that is specific to 3D symmetries was done by Mitra *et al.* [Mit+12]. Both these works cover a majority of the published papers from both the computer graphics and computer vision community in the topic of symmetry.

Hermann Weyl, in his book [Wey15] discusses the complexities associated with a general definition of symmetry and we summarize it here as follows: "*An object is said to have symmetry if some movement of the object or operation on the object leaves it in a position that is indistinguishable from its original position*". It is important to note that in order to define symmetry mathematically, we have to define what we mean by each of these terms: "object", "movement", "operation", "indistinguishable" and "position". Intuitive as these terms might seem to a reader, it is however not universal and it can have wide range of meanings in different applications like Chemistry, Physics, Biology and Engineering and in my own thesis.

In the most general case, an "object" is defined as a set on which we can define other properties that are essential to the definition of symmetry such as "position", "movement" and the notion of similarity which leads us to identify if the object is truly "indistinguishable"

from another. When we say "object" our minds automatically jump to types of everyday things that we interact with in the familiar 3D Euclidean space that we live in - like chairs, buildings, paintings *etc.*, all of which may exhibit some kind of symmetry as we intuitively understand it. This collection also encompasses the set of objects that we cannot see with the naked eye, like atoms, molecules, DNA etc. which however still reside in the 3D Euclidean space. Watson and Crick [Wat12] famously discovered the double helix structure of the DNA for which they were awarded the Nobel Prize in medicine in 1962. However symmetries are present beyond the familiar 3D Euclidean space that we occupy. Physicists have explored symmetries beyond the 3 dimensions and this has lead to discoveries of many Nobel prize winning ideas in that field and the concept of "object" in these dimensions is quite abstract and cannot be interacted with as we understand it intuitively. When we "see" symmetries in photographs and paintings, we are talking about its symmetric properties in an abstract two dimensional space that we cannot, again physically interact with because in reality there are no 2D objects in our 3D world. When mathematicians talk about symmetry, they often refer to infinite, abstract quantities such as *mathematical spaces* like the n-D Euclidean space $\mathbb{R}^n$ which is in fact symmetric. Such a wide range of meaning for a simple entity like "object" requires us to be very specific about our definition and yet general enough that it can be translated to a wide enough setting.

A "position" of an object can be intuitively viewed as it location in 3D space and location is strongly tied to the definition of distance, a scalar quantity between elements of an object (which has been defined earlier to be an abstract set). We can then say that two elements of a set have a unique location if the distance between its elements is non zero. If an object is defined as a set, the position of an object, is defined by the collective positions of its

individual elements. For an object in a Euclidean coordinate frame defined by a set of points, its location is defined by the position vector of each of the points and the distance can be defined by the Euclidean distance between the points. In a non-conventional, but a possible example, an object can be represented by a set of surface patches. Each surface patch consists of a finite set of points. We can then define a distance as the pairwise distance between the points of each patch. In a more abstruse example, an object could be CAD models of a building, let us say a set that consists of elements consisting of geometric entities like points, lines and plane segments. In such a case, we need to define the notion of distance (a scalar) between every pair of elements of a set such as point-line, point-point, line-point, line-line, plane segment-line, plane segment -point, plane segment-plane segment in order to define a location for the object. In general, defining distances for a homogeneous set is preferred since we do not have to normalize for different types of distances that arise in a heterogeneous set and it is therefore preferable to treat all the above entities (lines, planes etc.) as sets of points themselves.

"Movement/Operation", in its most general form is defined as a mapping of an object and "Indistinguishable" here refers to a state of the object such that the distance between the object before/after a movement is zero. For an object consisting of a set of points in a Euclidean coordinate frame, a movement transforms the position of an object specified by its coordinates (Cartesian or otherwise) or it could refer to the transformation of an entire space (For example movement of infinite plane in $\mathbb{R}^3$, which is an infinite set of points). If an object is defined as set and its position is defined by the collective positions of its individual elements, the distance is defined by the sum of the pairwise distance between the corresponding elements of the set assuming that every set has a corresponding element

in the other set. In such a case, "Indistinguishable" is a state where the distance between the sets is zero.

The goal of the above introduction to symmetry was to demonstrate the difficulty associated with providing a general definition of symmetry that is applicable across a different set of domains. Even within the computer vision community, there is a lack of consensus on its definition and the variability usually depends on its application to images, point clouds, 3D models, surfaces etc. Now that I have described the kind of challenges that we face while defining symmetry, I will provide a rigorous mathematical definition for it in the next section.

Definition

In geometry, the definition of symmetry is rooted in the definition of **isometry**. Martin and George [Mar12] define isometry for a **set of points** $\mathcal{X}$ in a Euclidean space as follows

---

Definition 1 (1)**.** *An isometry* P *is a mapping from a set of points* $\mathcal{X}$ *into itself that preserves distance. The set of all isometries* P *of* $\mathcal{X}$ *is denoted by* $\mathcal{P}_{\mathcal{X}}$*. If,* $\mathsf{P} : \mathsf{x} \mapsto \mathsf{x}\prime$*, then* P *is an isometry* ***iff*** $\mathsf{x}_1\prime\mathsf{x}_2\prime = \mathsf{x}_1\mathsf{x}_2$ *for all points* x_1 *and* x_2*.*

The notion of isometry is a general idea that is not restricted to Euclidean spaces or point sets within mathematics. In fact, the word isometry means "preserving distances". The word "metric" is a synonym to the word "distance". While Definition 1 establishes the notion of isometry for point sets in a Euclidean space, it can be defined on points that could exist in non-Euclidean spaces, provided we define the notion of distance correctly for these sets. For example Khan *et al.* [Kha12] describe isometries of points in hyperbolic spaces. While we only deal with symmetries of sets of points ("objects") in this thesis, isometry can be defined for other geometric entities like lines [Mar12] and even 'non-geometric' objects like pixels. While in most cases it is trivial to decompose these into points (a line is an infinite set of points and a pixel can be seen as location in 2D $(\in \mathbb{R}^2)$ along with intensity $(\in \mathbb{R})$, effectively making it a point in $\mathbb{R}^3$). However when such kinds of decompositions are not desirable or not possible, we can directly define isometry on these "objects" without having to predefine isometries on points. For example, Martin *et al.* [Mar12] show that reflections with respect to a plane that is perpendicular to a line is an isometry of a line. This can be justified as follows: if the line is an infinite set of points, reflection along this plane leaves the pairwise Euclidean distance between two points unchanged and is hence an isometry. If the line is infinite, it is trivial to see that translations parallel to the line also does not change the pairwise distance between two points and is also an isometry. Similarly, an infinite plane can also have translations parallel to the plane and reflection along another plane that is perpendicular to the plane under consideration, as potential isometries. In addition, a plane can also be rotated around an axis that is perpendicular to the plane and still leave it unchanged.

Burago *et al.* [BBI01] extends the definition of isometry to a general setting as follows.

Let X and Y be "metric spaces" with metrics (a.k.a. distances) d_X and d_Y, isometry can be defined as follows:

Definition 2 (2)**.** *A map* $\mathsf{P} : X \to Y$ *is called an isometry if for any* $\mathsf{x}_1, \mathsf{x}_2 \in X$ *one has* $d_Y\left(\mathsf{P}\left(\mathsf{x}_1\right), \mathsf{P}\left(\mathsf{x}_2\right)\right) = d_X\left(\mathsf{x}_1, \mathsf{x}_2\right)$

---

where a metric space is a set in which pairwise distances between the set elements is defined for *every* pair and an isometry preserves this distance. However it is to be noted that the above argument is not essential for understanding the claims of this thesis and was an aside to an interested reader thinking about symmetries in projective spaces as Liu *et al.* [Liu+10] briefly discusses in her survey [Liu+10].

Given that we have provided a concrete definition of isometry, we can now provide the following definition of **symmetry**:

Definition 3 (3)**.** *If* $\mathcal{X}$ *is a set of points such that for all its elements, distance* d_X *between any pair of its elements is defined, An isometry* S *is a symmetry of* $\mathcal{X}$ *if and only if* $\mathsf{S}\left(\mathcal{X}\right) = \mathcal{X}$*. Conversely, a set* $\mathcal{X}$ *is said to be symmetric if there is any isometry* S *such that* $\mathsf{S}\left(\mathcal{X}\right) = \mathcal{X}$*.*

All symmetries are isometries but the converse is not true. For instance all Euclidean transformations are isometries but not all are symmetries. While the above definition of symmetry is geometric, it is also essential to understand symmetry from a group theoretic

point of view. **Groups** have many important properties and allow us to compactly classify a wide range of symmetries and more importantly allow us to guess at symmetries which are present but not immediately obvious (geometrically). Relying on geometry alone to recover symmetries while dealing with noisy data that is suffering from a host of problems like occlusions can often be impossible unless we have a good initial guess. In the reminder of this section we briefly talk about groups and define important properties associated with groups. We mainly use Roy Mcweeny [McW02] who goes into some detail about the group theoretic stand-point on symmetry.

Definition 4 (4)**.** *Let* $\mathcal{G} = \{\mathsf{G}_1, \mathsf{G}_2 \ldots\}$ *be a non-empty set with a well-defined binary operation* $\circ$ *such that for each ordered pair* $\mathsf{G}_1, \mathsf{G}_2 \in \mathcal{G}$ $\mathsf{G}_1 \circ \mathsf{G}_2$ *is also in* $\mathcal{G}$*.* $(\mathcal{G}, \circ)$ *is a group if and only if:*

---

1. there exists an identity element $\mathsf{I} \in \mathcal{G}$ such that $\mathsf{I} \circ \mathsf{G} = \mathsf{G} = \mathsf{G} \circ \mathsf{I}$ for all $\mathsf{G} \in \mathcal{G}$.
2. any element G in $\mathcal{G}$ has an inverse $\mathsf{G}^{-1} \in \mathcal{G}$ such that $\mathsf{G} \circ \mathsf{G}^{-1} = \mathsf{G}^{-1} \circ \mathsf{G} = \mathsf{I}$.
3. the binary operation $\circ$ is associative: $\mathsf{G}_1 \circ (\mathsf{G}_2 \circ \mathsf{G}_3) = (\mathsf{G}_1 \circ \mathsf{G}_2) \circ \mathsf{G}_3$ for all $\mathsf{G}_1, \mathsf{G}_2, \mathsf{G}_3 \in \mathcal{G}$.

Using the composition of mappings in $\mathbb{R}^n$ as the binary operation $\circ$, one can prove that symmetries of a subset $\mathcal{X} \subset \mathbb{R}^n$ form a group, called the **symmetry group** $\mathcal{G}_\mathsf{S}$ of $\mathcal{X}$ [Liu+10]. It is important to note the symmetry groups are a set of symmetries (Definition 3) operating either separately or in combination on an object. In other words the set $\mathcal{X}$ is said to

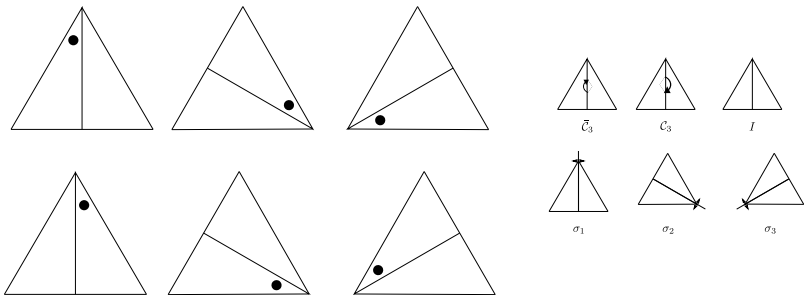


Figure 2.1: **System of Generators and Action of a Group on a Set**: Shown above is the group C_{3v}. The group has 5 elements $\mathcal{G}_{3v} = \left\{I_2, \mathcal{C}_3, \bar{\mathcal{C}}_3, \sigma_1, \sigma_2, \sigma_3\right\}$ where I_2 is the identity element, $\mathcal{C}_3, \bar{\mathcal{C}}_3$ are clockwise and counterclockwise rotations of 120^0 about the center, $\sigma_1, \sigma_2, \sigma_3$ are reflection with respect to one of the 3 lines. The group elements generate six distinct configurations shown on the left. Any combination of them can only generate one of these 6 configurations, for example $\sigma_1^{100}\mathcal{C}_3^{20} = \bar{\mathcal{C}}_3$ where superscript indicates the number of times the operation is applied. There are many system of generators for the group, for example $\mathsf{g}_1 = \mathcal{C}_3$ and $\mathsf{g}_2 = \sigma_2$ can generate all elements of the group. But this is not the only system, $\mathsf{g}_1 = \bar{\mathcal{C}}_3$ and $\mathsf{g}_2 = \sigma_2$ can also generate all the elements of the group.

be symmetric under all the operations defined in $\mathcal{G}_\mathsf{S}$ and any compositions of them applied infinitely. Figure 2.1 illustrates this with an example.

To better understand the action of a group on a set $\mathcal{X}$, Müller *et al.* [Mül13] defines the following set of properties that define the action of a group on a set:

Definition 5 (5)**.** *The group $\mathcal{G}_\mathsf{S}$ acts on the set $\mathcal{X}$ if:*

1. $\mathsf{x}_j = \mathsf{g}_k\mathsf{x}$ *is a unique element* $\forall \mathsf{x}, \mathsf{x}_j \in \mathcal{X}, \mathsf{g}_k \in \mathcal{G}_\mathsf{S}$
2. $I_N\mathsf{x} = \mathsf{x}$ *and* $I_N \in \mathcal{G}_\mathsf{S}$
3. $\mathsf{g}_k\left(\mathsf{g}_i\mathsf{x}\right) = \left(\mathsf{g}_k\mathsf{g}_i\mathsf{x}\right)\mathsf{x}$ *holds for every pair* $\mathsf{g}_k, \mathsf{g}_i \in \mathcal{G}_\mathsf{S}$

Since the group potentially generates an infinite number of symmetries for an object, we need to define the **minimum** number of elements of the set that fully define a symmetry group $\mathcal{G}_\mathsf{S}$. We call this minimum set as a **system of generators.** Conversely, all the elements of a group can be expressed as products whose factors are drawn from proper subset of the elements of the system of generators. Mcweeny [McW02] defines generators of a group as follows:

Definition 6 (6)**.** *A **system of generators** of a group $\mathcal{G}$ is a subset such that, every element of $\mathcal{G}$ can be written as the "product" of a finite number of factors, each of which is either an element of the subset or the inverse of such an element. Every element of the subset is called a **generator** g of the group $\mathcal{G}$.*

Generators are a compact way of representing a wide range of symmetries and this is a convenient parameterization for **generative models,** one of the important representation of 3D shapes that we seek to identify in this thesis. While Definition 5 provides a one-way relation between a group and its action on an element (a set of points), we can also define another entity that describes the relationship between the points of a symmetric set $\mathsf{x} \in \mathcal{X}$. This is important because it leads us to a realization that not all points in a symmetric set are equivalent and some points exhibit a *higher degree of symmetry* than others in the set. This notion is captured by the term **orbit:**

Definition 7 (7)**.** *The set of elements $\mathsf{x}_j \in \mathcal{X}$ which are obtained by $\mathsf{x}_j = \mathsf{g}_k\mathsf{x}$ when g_k runs through all elements of the group $\mathcal{G}_\mathsf{S}$ is called the $\mathcal{G}_\mathsf{S} - orbit$ of x or the orbit $\mathcal{G}_{\mathsf{S}\mathsf{x}}$.*

Another related term to orbits that operates on a subset of elements of a group called **stabilizer**

Definition 8 (8)**.** *The set of all* $\mathsf{g}_i \in \mathcal{G}_\mathsf{S}$ *for which* $\mathsf{g}_i\mathsf{x} = \mathsf{x}$ *holds is called the stabilizer* S *of* x *in* $\mathcal{G}_\mathsf{S}$

By using the definition of stabilizers we can define the term orbit of a point in a symmetric set that exhibits translational symmetry by forming a stabilizer of the group that excludes the translational symmetry. We can now define a quantifiable entity called **length of an orbit of a stabilizer** or the **order of the stabilizer,** that define the order of the symmetry at a point $\mathsf{x} \in \mathcal{X}$. This is also called the site symmetry of a point and we will revisit this later in the chapter.

Before we proceed further we need to define two more terms **representation** and **isomorphism**. Symmetries in 3D, such as the types that we are interested in, is essentially a transformation, which can be represented in variety of formats. For instance rotations can be represented by axis-angle, Euler rotations, rotation matrices, quaternions etc. each of which is distinct mathematical entity that has a well defined domain. The specific manifestation of the elements of the group is called a representation. When we choose a specific representation, it is required that we choose the same representation for all elements of the set of generators g such that composition operation $\circ$ is defined on any pairs of elements in P. All the other alternate representations of g form a new group that is isomorphic to each of the other representations. Mcweeny [McW02] formally defines isomorphism as follows:

Definition 9 (9)**.** *A mapping of* $(G, \circ)$ *to* $\left(\bar{G}, \star\right)$ *is called isomorphic when the condition* $(R \circ S)\prime = R\prime \star S\prime \ \forall \ R, S \in G, R\prime S\prime \in \bar{G}$ *such that* $\bar{G}$ *is an image of* G *with a one-to-one and unique mapping* $R \rightarrow R\prime$*.*

Isomorphism allows to represent symmetries by wide set of representations, which is rather important when it comes to optimization. This is because choice of representations determine the complexity and robustness of the optimization process to noise and other parameters.

Lattices

Determining symmetry alone is not sufficient to address all the objectives of my thesis. One of the primary goal of my thesis is to extract meaningful sets of points called "basis elements". Symmetry, while providing a pairwise relationship between points in the form of Euclidean transformations, does not provide us with a meaningful group of points that belong together. In other words, determining the symmetries of a scene does not provide a segmentation of the scene into its symmetric elements. We therefore need a separate mathematical machinery in place to solve the problem of grouping likened points and extracting basis elements.

Lattices, a popular mathematical and geometric construct used in crystallographic theory provides the necessary framework for solving the problem of grouping similar points (a.k.a segmentation) and basis element extraction. Lattices complement the definition of symmetry, and together, they fully define a generative model of the 3D geometry and allow us to provide **compressed**, **semantically meaningful** point clouds that is one of the major objectives of this thesis.

In this thesis, it will be useful to classify symmetry into two subcategories: symmetries that (1) exist in a lattice and (2) do not exist in a lattice. In order to understand this classification, we first need to define a **lattice**. The idea of lattices has been discussed by many prominent mathematicians and, similar to symmetry, they also exhibit group structure [CS13]. I will summarize some of the important concepts and definitions here. For a detailed explanation of these, refer to Müller's book on crystallography [Mül13] from which we have borrowed most of the definitions.

Definition

A **vector lattice** in $\mathbb{R}^n$ is defined by Müller *et al.* [Mül13] as follows

---

Definition 10 (10)**.** *For any basis of* $\mathbb{R}^n$*, the subgroup of all linear combinations with integer coefficients of basis vectors* b *forms a vector lattice.*

This can be expressed mathematically as the infinite set of vectors such that an element t of the set can be generated as

$$\mathsf{t} = \sum_{i=1}^{n} q_i \mathsf{b}_i \quad \forall q_i \in \mathbb{Z} \tag{2.1}$$

$$= \mathsf{B}\,\mathsf{q} \tag{2.2}$$

where b_i's are the basis vectors that span $\mathbb{R}^n$ and B is the basis matrix whose i^{th} column corresponds to the basis vector b_i. $\mathsf{q} = [q_1 \dots q_n] \,\big|\, q_i \in \mathbb{Z}$ are called the **Miller indices.** These indices only allow integer combination of basis vectors. In crystal lattices, the basis vectors b are called **translational vectors** [Mül13] where a translational vector is defined

as a shift that brings a crystal into superposition with itself. A vector lattice in $\mathbb{R}^n$ can be represented by $2n$ parameters that corresponds to magnitudes of the n basis vectors and the n pairwise angles between them.

The Metric Tensor

The basis matrix B is an over-parametrized representation of lattices (requiring n^2 entries) while the **metric tensor** $\mathbf{G}$ is much more compact representation of vector lattices (requiring at most $2n$ entries). For example, the metric tensor corresponding to a vector lattice consisting of magnitudes $[a, b, c]$ and angles $[\alpha, \beta, \gamma]$ in $\mathbb{R}^3$ is given by

$$\mathbf{G} = \begin{bmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & g_{33} \end{bmatrix} = \begin{bmatrix} a^2 & ab\cos\gamma & ac\cos\beta \\ ab\cos\gamma & b^2 & bc\cos\alpha \\ ac\cos\beta & bc\cos\alpha & c^2 \end{bmatrix} \tag{2.3}$$

$|\mathbf{G}|$ corresponds to the volume of the parallelepiped formed by the three translational vectors of the lattice in $\mathbb{R}^3$. We can extend this analogy in $\mathbb{R}^n$ and show that the metric tensor in $\mathbb{R}^n$ corresponds to an $n \times n$ matrix whose determinant is the volume of the $n-$dimensional parallelotope.

Basis Matrix $\leftrightarrow$ Metric Tensor We have defined two forms of representing lattices and we now see how they can be converted from one form to another. Consider the case of a 3D lattice specified by magnitudes of vectors $[a, b, c]$ and angles $[\alpha, \beta, \gamma]$ where α is the angle between b_1 and b_2, β, between b_0 and b_2 and γ, between b_0 and b_1. Equation 2.3 can now directly provide the metric tensor. Now how do we arrive at the basis matrix from this

representation? Since we are free to choose any coordinate system for the lattice, without any loss of generality, we can pick the first basis vector b_0 as $\begin{bmatrix} a & 0 & 0 \end{bmatrix}^T$ which is a vector along the x-axis with a magnitude of a. Similarly b_1 is any vector that is at an angle γ to b_0 with a magnitude of b we can therefore choose this as $\begin{bmatrix} b\cos(\gamma) & b\sin(\gamma) & 0 \end{bmatrix}^T$. b_2 is at angle of $\alpha$to b_0 and β to b_1. We can obtain b_2 by solving the constraints generated by the pairwise products between the vectors. For example $\mathsf{b}_0 \cdot \mathsf{b}_2 = b \cdot c \cdot \cos(\beta)$ therefore, the first component of the vector is $c \cdot \cos(\beta)$. Similarly we can get the other components of the vector by solving for the constraints between b_0 and b_1 and the third component can be obtained by the constraint that the magnitude of the vector is known $\|\mathsf{b}_2\| = c$. We can therefore convert between the two representations and and each representation has a unique advantage for analyzing certain properties of the lattice. For example, the metric tensor is useful for obtaining the distance of a point in a lattice from its origin or any other point directly from lattice coordinates while the basis matrix requires us to individually compute the coordinate of the two points and then compute the distance between them.

The Unit Cell

In order to specify a location in a lattice, we need to define another term called **unit cell**. Defining the unit cell serves two purpose: (1) it allows us to define the location and orientation of the lattice with respect to an external coordinate frame and (2) it lays the foundation for defining the basis element that we will discuss in Chapter 6. The unit cell is formally defined by Müller *et al.* [Mül13] as

---

Definition 11 (11)**.** *In $\mathbb{R}^n$, the parallelotope in which the coordinates of all points within the parallelotope are $0 \leq \mathsf{u}\prime_1, \mathsf{u}\prime_2, \mathsf{u}\prime_3 \ldots \mathsf{u}\prime_n < 1$ is called a unit cell.*

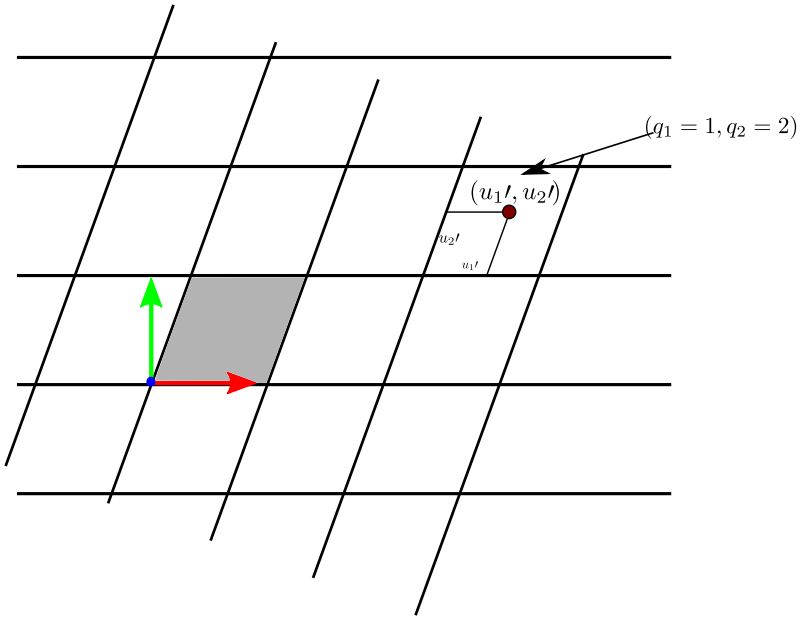


Figure 2.2: **Location of a point in a lattice**: Given that we have defined the origin (blue) and the unit cell (gray), the location of a point x in this 2D lattice is given by the Miller indices q and its location inside the unit cell $\mathsf{u}\prime$. In this case the origin of the lattice coincides with the world origin ${}_w\mathsf{t}_l = \mathbf{0}$ and the basis vectors b are expressed in the world's coordinate frame.

We call any point that is inside the unit cell as a **primary point**. Similar to symmetry lattices exhibit group structure. A lattice in $\mathbb{R}^n$ is a symmetry group of discrete translational symmetry in n directions and it is isomorphic to the group of integers $\mathbb{Z}^n$.

Location of a point in a lattice

The location of a point x in a lattice in $\mathbb{R}^n$ is fully expressed by its Miller indices $\mathsf{q} \in \mathbb{Z}^n$ and its coordinates inside a unit cell $\mathbf{0} \leq \mathsf{u}\prime \leq \mathbf{1}$, and a known location of the origin of the lattice ${}_w\mathsf{t}_l$ as:

$$\mathsf{x} =_w \mathsf{t}_l + \mathsf{B}\ (\mathsf{u}\prime + \mathsf{q}) \tag{2.4}$$

where $\mathsf{B} \in \mathbb{R}^{n \times n}$ is the basis matrix whose columns are the basis vectors b and ${}_w\mathsf{t}_l \in \mathbb{R}^n$ is the location of the origin of the lattice. Figure 2.2 shows the geometry of a 2D point in a 2D non-orthogonal lattice. The basis matrix B is a 2×2 matrix, the location of the point inside the unit cell is given $\mathbf{0^2} \leq \mathsf{u\prime} \leq \mathbf{1}^2$ and its Miller indices $\mathsf{q} = \begin{bmatrix} 1 & 2 \end{bmatrix}^T$. Given that the basis vectors bare expressed in the coordinate system of the in which the point ${}_w\mathsf{t}_l$ is given, we can see that Equation 2.4 follows directly. We can extend the same analogy to any n-D point in an n-D lattice

However when it comes to architectural data, the dimensionality of the lattice is usually less the dimensionality of a point $\mathsf{x} \in \mathbb{R}^3$. It is therefore imperative that we understand the significance of the basis matrix for **sub-dimensional lattices** and how we can express Equation 2.4 for these scenarios. We will explore cases of sub-dimensional lattices that commonly occur in urban scenes and then generalize this notion in the subsequent section.

3D point in a 2D lattice In this case: we want to express a 3D point $\mathsf{x} \in \mathbb{R}^3$ in a 2D lattice specified by $\mathsf{u\prime\prime} \in \mathbb{R}^2$. The location of the point x in the lattice is specified by its Miller indices $\mathsf{q} \in \mathbb{Z}^2$ and its location inside the unit cell $\mathsf{u\prime} = [\mathsf{u\prime}_1, \mathsf{u\prime}_2]^T \; \forall \; \mathbf{0} \leq \mathsf{u\prime} \leq \mathbf{1}$. Such a kind of configuration commonly occur on building facades that exhibit 2D symmetries in the form of recurring patterns of windows. In this case the columns of the basis matrix $\mathsf{B} \in \mathbb{R}^{2 \times 2}$, i.e.the basis vectors b are 2D vectors specified in the local coordinate frame of the lattice. Equation 2.4 can now be written by the following dimensionally incorrect equation:

$$\mathsf{x} =_w \mathsf{R}_l \cdot \underbrace{\mathsf{B} \cdot (\mathsf{u\prime} + \mathsf{q})}_{2d\,point} +_w \mathsf{t}_l \tag{2.5}$$

which is similar to Equation 2.4 except that we specify the full *pose of the lattice* as $({}_w\mathsf{R}_l \in \mathbb{SO}(3), {}_w\mathsf{t}_l \in \mathbb{R}^3)$ instead of its location ${}_w\mathsf{t}_l$ in space as specified in its trivial counterpart, Equation 2.4. In fact it is trivial to notice that the Equation 2.4 is a special case of Equation 2.5. By decoupling the action of rotation (the orientation of the lattice with respect to the world origin) while expressing B, we are free to choose our own coordinate system while representing points *inside* the lattice, possibly even non-orthogonal ones as shown in Figure 2.2 and exploit the well developed theories around 2D lattices and its associated properties.

However, there is a **dimensional inconsistency** in the above equation: $B \in \mathbb{R}^{2\times2}$ and $R_0 \subset \mathbb{R}^{3\times3}$. In order to solve this problem we need to consider the geometry of a sub-dimensional lattice shown in Figure 2.3 and the meaning of identity rotation matrix in Equation 2.5This can be understood by considering the following example: Let us consider the case when the lattice is parallel to the $x-y$ plane with the origin of the lattice is located on the $z-$axis. For this case ${}_w\mathsf{t}_l = \mathbf{0}$ and ${}_w\mathsf{R}_l = I_3$ **iff** the 2-D point in the lattice $\mathsf{u}\prime\prime \in \left([0,1]^2, \mathbb{Z}^2\right)$ is augmented by its depth value z to express it in a location in $\mathbb{R}^3$:

$$\mathsf{x} = \begin{bmatrix} \mathsf{B}\cdot(\mathsf{u}\prime + \mathsf{q}) \\ \mathsf{v}\prime \end{bmatrix} \tag{2.6}$$

where $\mathsf{x} \in \mathbb{R}^3$. If we allow that the lattice can be rotated and translated from this "identity" configuration i.e.${}_w\mathsf{T}_l = ({}_w\mathsf{R}_l, {}_w\mathsf{t}_l) \neq I_4$, then, we transform the point into the world coordinate frame as

$$\mathsf{x}_w = {}_w\mathsf{R}_l\mathsf{x}_l + {}_w\mathsf{t}_l$$

where x_l is specified by Equation 2.6 with z as the "depth" of the point with respect to the plane of the lattice. If the point is on the lattice(as in this example $z = 0$, but this will not be the case for 3D windows in architecture which have a sense of depth associated with them) x_w is the coordinate in the world coordinate frame. We can now combine the above two steps to give a single equation for converting between the position of a point in 2D lattice into its 3D locations as

$$\mathsf{x} =_w \mathsf{R}_l \cdot \begin{bmatrix} \mathsf{B}\left(\mathsf{u}\prime + \mathsf{q}\right) \\ \mathsf{v}\prime \end{bmatrix} +_w \mathsf{t}_l$$

where the $\mathsf{v}\prime$ is the depth of the point in the lattice with respect to lattice's 2D plane.

3D point in a 1D lattice In this scenario, $\mathsf{u}\prime\prime \in \left([0,1], \mathbb{Z}\right)$ and ${}_w\mathsf{T}_l = \left({}_w\mathsf{R}_l, {}_w\mathsf{t}_l\right) \in \mathbb{SE}\left(3\right)$, $\mathsf{x} \in \mathbb{R}^3$ and $\mathsf{B} \in \mathbb{R}$. And most importantly $\mathsf{v}\prime$ is given by the vector that specifies the projection onto any two perpendicular vectors that are also perpendicular to 3D line, parallel to the $x-$ axis of the world coordinate frame and passing through the origin of the lattice. In this setting the "identity rotation" configuration is that in which the other two perpendicular axes are chosen parallel to the y and z axis of the world coordinate frame. The rotation matrix now has two degrees of freedom as compared to the previous case.

$N-$D point in a $L-$D lattice We can generalize the above case to $N-$dimensional point in a $L-$dimensional lattice $\forall N \leq L$ as the following interrelation between the same point x specified in lattice space $\mathbb{L}_{N,L} = \left(\mathbb{SE}\left(N\right), \left(\left[0,1\right]^L, \mathbb{Z}^L\right)\right)$ to its location in $\mathbb{R}^N$ as

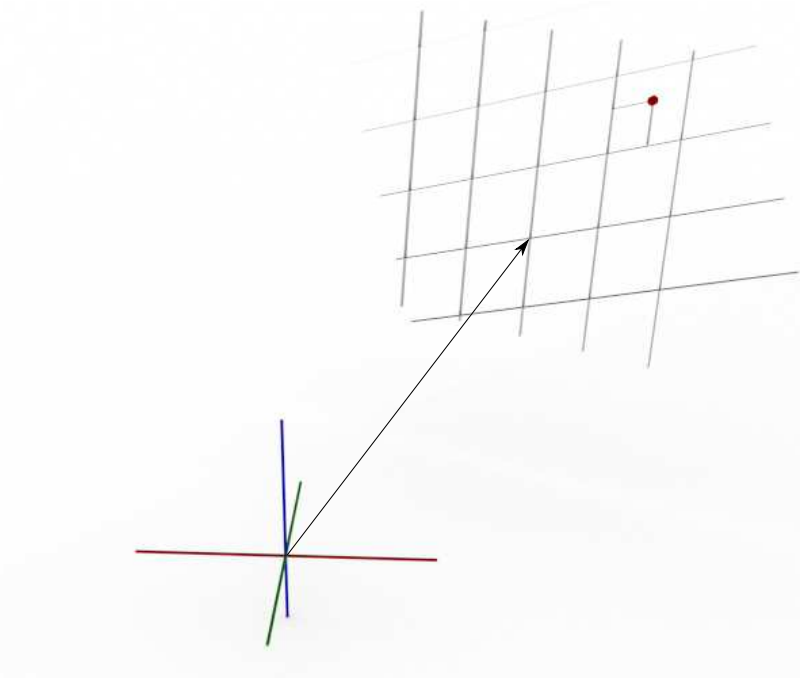


Figure 2.3: **Sub-dimensional lattices**: The 2D lattice shown in Figure 2.2 now represented in a 3D coordinate system with the origin not coinciding with the origin of the lattice. A point in a sub-dimensional lattice $\mathsf{x} \in \mathbb{L}_{3,2}$where $\mathbb{L}_{3,2} = \left(\mathbb{SE}\left(3\right), \left(\left[0,1\right]^2, \mathbb{Z}^2\right)\right)$ can be converted to a point in the global coordinate frame in 3D $(\mathbb{R}^3)$ by using Equation 2.7 with $N = 3$ & $L = 2$.

follows:

$$\mathsf{x} =_w \mathsf{R}_l \begin{bmatrix} \mathsf{B} \cdot (\mathsf{u}\prime + \mathsf{q}) \\ \mathsf{v}\prime \end{bmatrix} +_w \mathsf{t}_l \tag{2.7}$$

where $\mathsf{x} \in \mathbb{R}^N$, $({}_w\mathsf{R}_l \in \mathbb{R}^{N\times N}$, ${}_w\mathsf{t}_l \in \mathbb{R}^N)$ specifies the pose of the $L-$dimensional lattice in a $L-$dimensional space. the basis matrix $\mathsf{B} \in \mathbb{R}^{N\times N}$ defines the lattice.Importantly, $\mathsf{v}\prime \in \mathbb{R}^{N-L}$ is the vector that specifies the "depth vector" onto the $N - L$ perpendicular axes that are are also perpendicular to the k mutually perpendicular axes that specifies the subspace of the $L-$dimensional lattice in $\mathbb{R}^N$. The identity rotation is that configuration

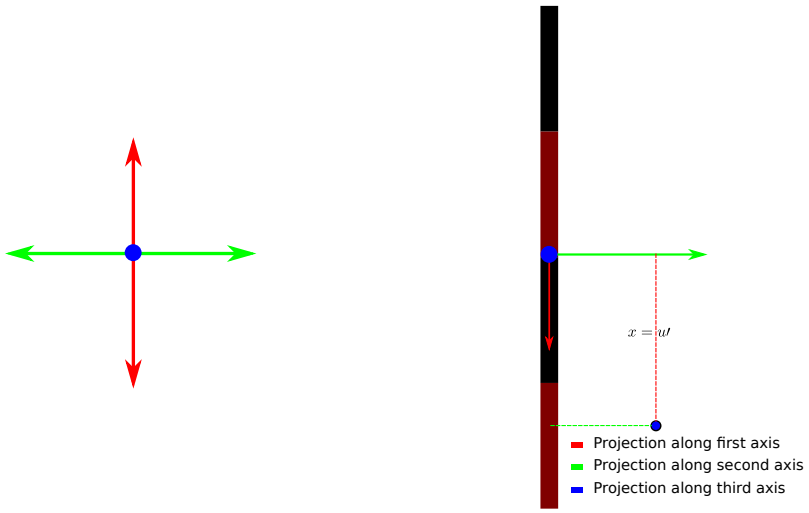


Figure 2.4: **Sub-dimensional lattices** II: 1D lattice in a 3D coordinate frame with the origin not coinciding with the origin of the lattice. A point in a sub-dimensional lattice $\mathsf{x} \in \mathbb{L}_{3,1}$ (shown in black) where $\mathbb{L}_{3,1} = (\mathbb{SE}(3), ([0,1] \quad, \mathbb{Z}))$. $\mathsf{v}\prime \in \mathbb{R}^2$ in this case is obtained by projecting the point onto two mutually perpendicular directions that is also perpendicular to the direction of the lattice (that is chosen to be parallel to the world coordinate frame's x-axis). The identity rotation is defined as the case the when the chosen two directions are parallel to the y and z axis of the world coordinate frame.

when the all the axes are parallel to the axes of the 3D coordinate axes and it is tied to our choice in which we pick the mutually perpendicular axes of the lattice; a configuration could have a "simpler" simply because we pick the axis cleverly and we will go into the details of this in Chapter 6.

Fundamental Symmetry operations in Canonical Coordinate Frame

We have so far provided a means to describe the location of a point in a lattice by its location in the unit cell $\mathsf{u}\prime$ and its Miller indices q. However this does not address the symmetries that often occur within the unit cell itself. In order to understand the types of symmetry groups that can occur inside the unit cell, we need to first understand the

possible **symmetry operations (SO)** that can occur in $\mathbb{R}^N$. Muller *et al.* [Mül13] defines symmetry operation:

Definition 12 (12)**.** *A symmetry operation is a mapping of an object such that (1) all distances remain unchanged (2) the object is mapped onto itself or onto its mirror image. If the object is a crystal structure, the mapping is a crystallographic symmetry operation*

We can represent all mappings that occur in $\mathbb{R}^N \ \forall \ N = 1, 2, 3$ by an affine transformation, represented by an augmented affine matrix $\begin{bmatrix} \mathsf{W} & \mathsf{w} \\ 0 & 1 \end{bmatrix} \in \mathbb{R}^{(N+1)\times(N+1)}$ that operates on a homogeneous point in $\mathbb{R}^{N+1}$. In addition to this, since we are interested in affine mappings that are symmetries, the augmented matrix must also be an isometry in $\mathbb{R}^N$ and we can ensure that the determinant of the matrix $\det(\mathsf{W}) = \pm 1$. Therefore an an affine mapping in $\mathbb{R}^N$ is an isometry **iff**:

$$\begin{bmatrix} \mathsf{W} & \mathsf{w} \\ 0 & 1 \end{bmatrix} \forall \ \det(\mathsf{W}) = \pm 1$$

where $\mathsf{W} \in \mathbb{R}^{N\times N}$ and $\mathsf{w} \in \mathbb{R}^N$ for any symmetry operations in $\mathbb{R}^N$. There are 6 possible symmetry operations in $\mathbb{R}^3$ and a subset of those in $\mathbb{R}^2$ and $\mathbb{R}^1$. We enumerate these here:

1. **Translations**: There are isometries in which $\mathsf{W} = I_N$ and $\mathsf{w} \in \mathbb{R}^N$. They can occur in $\mathbb{R}^1, \mathbb{R}^2, \mathbb{R}^3$.

2. **Inversions:** Inversions point reflection isometries. Inversions through $(0, 0, 0)$ are represented by $\mathsf{W} = -I_N$ and $\mathsf{w} = 0$. Groups that have inversions are represented by the symbol $\bar{1}$. Inversions can occur in $\mathbb{R}^1, \mathbb{R}^2$ and $\mathbb{R}^3$. Although in 1D, inversions are same as reflections.

3. **Rotations**: They are isometries in which $\det(\mathsf{W}) = 1$ and $\mathsf{w} = \mathsf{0}$. They can occur in $\mathbb{R}^2, \mathbb{R}^3$;rotations in $\mathbb{R}^1$ is an identity. Rotations are denoted by a number that denotes its *order* when we represent groups that exhibit rotational symmetry $(1, 2, 4, \ldots)$.

4. **Reflections**: They are isometries in which $\det(\mathsf{W}) = -1$ and $\mathsf{w} = \mathsf{0}$. They can occur in $\mathbb{R}^1, \mathbb{R}^2, \mathbb{R}^3$. The presence of reflections in the groups is denoted by the letter m.

5. **Glide Reflections**: While m represents just a reflection along a line or a plane depending on the points in $1D, 2D$ or $3D$, glide reflections are more complex. The most commonly occurring glide reflections with respect to a glide plane that is parallel to one of the basis. In $2D$ the glide reflections are denoted by g and in 3D the are denoted by a, b, c depending on if the glide vector is one of the basis. n, if the glide vector is parallel t on the primary planes and d if the length of the glide vector is $1/4$ of the length of the basis. e is for glide planes having two mutually perpendicular glide vectors.

6. **RotoInversion**: They represent a rotation followed by an inversion. They are isometries in which $\det(\mathsf{W}) = -1$ and $\mathsf{w} = \mathsf{0}$. They are represented by the order of the rotation plus the inversion operator $\left(\bar{2}, \bar{3}, \bar{4} \ldots\right)$

7. **Screw Operation**: This is rotation followed by a translation parallel to the axis of rotation. they are represented by N_p where p/N is the shortest lattice dimension along the translation axis.

Yanxi *et al.* [Liu+10]calls these **primitive symmetries** and defines them as those symmetries that cannot be decomposed into simpler symmetries.

Asymmetric Unit

One of the primary goals of this thesis is to obtain the minimum number of points that can fully describe a symmetry structure. Given the points inside this minimal set, and the set of symmetry operations, we can fully describe the entire symmetric structure. Asymmetric units are therefore powerfully compact representations that encode the full detail of the symmetric structure leading to many desirable features such as semantic representation and compression.

Kastner *et al.* [Kas+00] defines an asymmetric unit as follows

---

Definition 13 (13)**.** *An **asymmetric unit** is the smallest unit of volume that contains all of the structural information and that by application of the symmetry operations can reproduce the (1) unit cell in case of a lattice and (2) the entire symmetric structure when there are no lattices.*

Site Symmetry and Wyckoff Positions

A unit cell often exhibits a number of symmetries which form a group. If we disregard the translational component of these symmetries (which is described using the lattice geometry), these symmetry groups tend to form point groups (which we define later; for now, point groups are symmetry groups that leave a single point invariant). Identifying and describing these point groups are typically useful in describing the entire symmetric pattern. Crystallographic notations combine the point group notations together with the possible

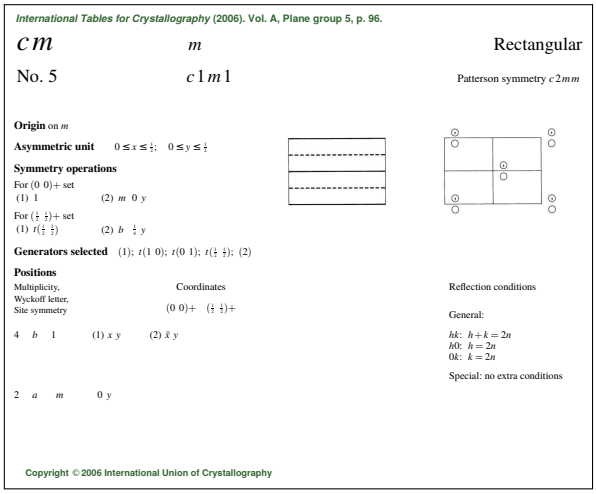
lattices to fully describe any possible symmetries in 3D. We now define two important concepts that help formalize symmetry inside the unit cells (1) **Site symmetry** and (2) **Wyckoff positions**

Definition 14 (14)**.** *The site-symmetry group (often called point symmetry) of a point is the finite group formed by the set of all symmetry operations of the space group of the crystal that leave that point invariant. It is isomorphic to a (proper or improper) subgroup of the point group to which the space group under consideration belongs. In general, the origin is a point of highest site symmetry.*

The above definition provides us with a unique definition of the **origin of the unit cell** as the center with the highest site symmetry. We showed that Miller Indices can be used to represent the location of a point in a lattice by means of its corresponding location within the unit cell. Similarly, crystallographers use the Wyckoff position to describe the location of a point in a unit cell with respect to its corresponding location in the asymmetric unit.

Definition 15 (15)**.** *In crystallography, a Wyckoff position is a point belonging to a set of points for which site symmetry groups are conjugate subgroups of the space group.[1] Crystallography tables give the Wyckoff positions for different space groups. There are two types of wyckoff positions (1) General positions and (2) Special positions.*

In contrast to Miller Indices that exclusively represent lattices, which are themselves a manifestation of translational symmetry alone, Wyckoff positions represent the order of



International Tables for Crystallography (2006). Vol. A, Plane group 5, p. 96.

cm m Rectangular

No. 5 $c\,1\,m\,1$ Patterson symmetry $c2mm$

Origin on m

Asymmetric unit $0 \le x \le \frac{1}{2}$; $0 \le y \le \frac{1}{4}$

Symmetry operations

For (0 0)+ set
(1) 1 (2) m 0 y

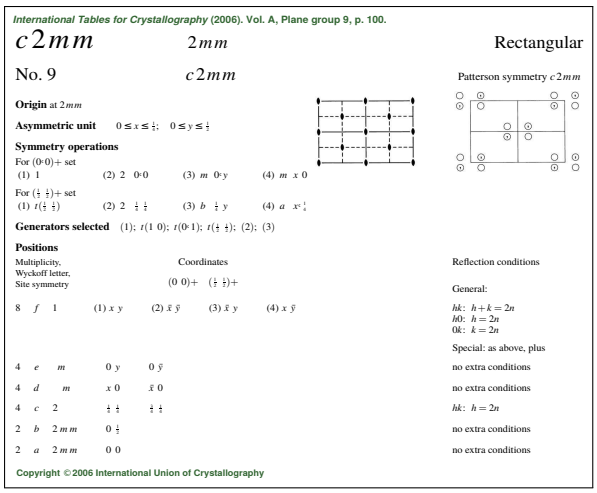
For $(\frac{1}{2}\ \frac{1}{2})$+ set
(1) $t(\frac{1}{2}\ \frac{1}{2})$ (2) b $\frac{1}{4}$ y

Generators selected (1); $t(1\ 0)$; $t(0\ 1)$; $t(\frac{1}{2}\ \frac{1}{2})$; (2)

Positions

Multiplicity, Wyckoff letter, Site symmetry	Coordinates $(0\ 0)+\ (\frac{1}{2}\ \frac{1}{2})+$		Reflection conditions
			General:
4 b 1	(1) $x\ y$	(2) $\bar{x}\ y$	hk: $h+k=2n$ $h0$: $h=2n$ $0k$: $k=2n$
			Special: no extra conditions
2 a m	0 y		

(a) Wallpaper group cm



International Tables for Crystallography (2006). Vol. A, Plane group 9, p. 100.

$c2mm$ $2mm$ Rectangular

No. 9 $c2mm$ Patterson symmetry $c2mm$

Origin at $2mm$

Asymmetric unit $0 \le x \le \frac{1}{4}$; $0 \le y \le \frac{1}{4}$

Symmetry operations

For (0·0)+ set
(1) 1 (2) 2 0·0 (3) m 0·y (4) m x 0

For $(\frac{1}{2}\ \frac{1}{2})$+ set
(1) $t(\frac{1}{2}\ \frac{1}{2})$ (2) 2 $\frac{1}{2}$ $\frac{1}{2}$ (3) b $\frac{1}{2}$ y (4) a x, $\frac{1}{4}$

Generators selected (1); $t(1\ 0)$; $t(0\ 1)$; $t(\frac{1}{2}\ \frac{1}{2})$; (2); (3)

Positions

Multiplicity, Wyckoff letter, Site symmetry	Coordinates $(0\ 0)+\ (\frac{1}{2}\ \frac{1}{2})+$				Reflection conditions
					General:
8 f 1	(1) $x\ y$	(2) $\bar{x}\ \bar{y}$	(3) $\bar{x}\ y$	(4) $x\ \bar{y}$	hk: $h+k=2n$ $h0$: $h=2n$ $0k$: $k=2n$
					Special: as above, plus
4 e m	0 y	0 $\bar{y}$			no extra conditions
4 d m	x 0	$\bar{x}$ 0			no extra conditions
4 c 2	$\frac{1}{4}$ $\frac{1}{4}$	$\frac{3}{4}$ $\frac{1}{4}$			hk: $h=2n$
2 b $2mm$	0 $\frac{1}{2}$				no extra conditions
2 a $2mm$	0 0				no extra conditions

(b) Layer group $c2mm$

Figure 2.5: **Representation of Wyckoff positions**: In this diagram we show the Wyckoff position of two symmetry groups (1) for the Wallpaper group and (2) for the Layer group.

non-linear symmetry operations (SO) such as the ones listed in Section 2.2 excluding translational symmetry. We already defined all the symmetry operations by using the general augmented affine matrix $\begin{bmatrix} \mathsf{W} & \mathsf{w} \\ 0 & 1 \end{bmatrix}$ By the property of isomorphism, any group consisting of one or more symmetry operation is isomorphic to the group that consists of augmented affine matrices (representing the symmetry operations) with the composition operation replaced by the matrix multiplication operator and the points represented as homogeneous coordinate vectors. Figure 2.5 shows an example of Wyckoff position represented in the international table of crystallography for a 2D wallpaper group(cm) and a 3D Layer group ($c2mm$). The international table specifies the positions inside the unit cell by the order of the site symmetry at each point . The first line represents the general position. This position has the lowest order(highest multiplicity) of site symmetry. In Figure 2.5 (a), we see that the multiplicity of the general position is 4 which means for

any point in the unit cell that is not at a special location, it has 4 symmetric manifestations that arise as a result of the symmetry operations in the space group of the unit cell. Two of these are listed in the table they are (1) x, y and (2) $-x, y$. We will describe how the other two points arise in the next section where we introduce lattice types. The second line in the table shows the site symmetry second lowest order and this corresponds to points in the reflection axis. This is a special position because unlike points in the general position, the points in special position have a lower multiplicity because they are invariant to atleast one symmetry operation, in this case it is the reflection operation. Similarly it case of 3D, the unit cell is a parallelepiped where the multiplicity of the general position is 8. This is because it has two orthogonal reflection planes. Despite the fact that this group has only two reflection planes we see a special position (marker c) that has a rotational symmetry of order 2 and multiplicity 4. This is because two perpendicular planes of reflection create a center of rotation that is of order 2. The crystallographic table represents all kinds of symmetry operations that occur inside the unit cell. We can choose a subset of these operations to generate all the points within the unit cell.

We now use the Wyckoff positions presented in the crystallographic table to provide a **formal mathematical framework for recreating the unit cell given points in the asymmetric unit** and an essential subset (also the system of generators of the points group inside the unit cell) of the Wyckoff positions. A recent work of Kohler *et al.* **[Koh+16]** provides a way to generate all the 17 wallpaper symmetries from a random asymmetric unit. In our gave we will show we can determine the 3D motifs given a particular symmetry pattern. The crystallographic table specifies the multiplicity of the different positions in the unit cell. The multiplicity of a point in the general position is a sum of

the multiplicity of the elements of the system of generators. Given a system of generators, $\{\mathsf{g}_1 \dots \mathsf{g}_L\}$ of a group $\mathcal{G}_{\mathsf{SU}}$ that is isomorphic to the set of augmented affine vectors $\left\{ \begin{bmatrix} \mathsf{W}_1 & \mathsf{w}_1 \\ 0 & 1 \end{bmatrix} \dots \begin{bmatrix} \mathsf{W}_L & \mathsf{w}_L \\ 0 & 1 \end{bmatrix} \right\}$, we can obtain a set of points in a unit cell that is symmetric with respect to the group $\mathcal{G}_{\mathsf{SU}}$ as

$$\mathsf{u}\prime = \left(\mathsf{g}_1^k \circ \dots \circ \mathsf{g}_L^l\right) \mathsf{u}$$

which is given by

$$\begin{bmatrix} \mathsf{u}\prime \\ 1 \end{bmatrix} = \begin{bmatrix} \mathsf{W}_1 & \mathsf{w}_1 \\ 0 & 1 \end{bmatrix}^k \dots \begin{bmatrix} \mathsf{W}_L & \mathsf{w}_L \\ 0 & 1 \end{bmatrix}^l \begin{bmatrix} \mathsf{u} \\ 1 \end{bmatrix} \tag{2.8}$$

Given a set of generators, we can now generate all the points inside the unit cell and the Wyckoff positions provide us the limit of the exponentiation of each generator. However not all lattices can be generated this way. Figure 2.5 (a)&(b), we can see that lattices are of the type centered lattices, hence denoted by the letter $'c'$ in the nomenclature. The Wyckoff positions, in addition to the site symmetry at location $(0,0)+$ also specify the translational symmetry at locations $\left(\frac{1}{2},\frac{1}{2}\right)+$. As a result, we will get another set of points in the unit cell as

$$\begin{bmatrix} \mathsf{u}\prime \\ 1 \end{bmatrix} = \begin{bmatrix} \mathsf{W}_1 & \mathsf{w}_1 \\ 0 & 1 \end{bmatrix}^k \dots \begin{bmatrix} \mathsf{W}_L & \mathsf{w}_L \\ 0 & 1 \end{bmatrix}^l \begin{bmatrix} \mathsf{u} + \frac{1}{2} \\ 1 \end{bmatrix} \tag{2.9}$$

In the next chapter, we demonstrate how the local choice of coordinate frames affects the local generators and why some representations are advantageous to optimization in comparison with others.

Symmetry classification using Groups and Lattices

The group structure exhibited by symmetries allow for many unique classifications. In this section we provide a means to classify these existing well-established symmetry groups using lattices that we defined in the previous section. This will allow us to later analyze and categorize the existing literature in terms of this classification and pinpoint our contribution in this field.

Lattice and Non-Lattice Symmetries

From a practical standpoint, it is useful to analyze translational symmetries in a separate framework; the lattice framework. There are several reasons for this: (1) translation symmetries are the equivalent of carbon atoms in chemistry, that readily combine with other kinds of symmetries to produce complex symmetries (2) they are abundant in the urban world compared to all other symmetries and (3) if analyzed separately they have several linear properties that other symmetries do not exhibit.

To conform with the prevailing segregation of symmetries in the symmetry literature, we also classify 3D symmetries into lattice and non-lattice symmetries and further provide a fine grained classification in terms of 1D, 2D and 3D lattices. Lattices have been explored in computer vision although, their definition is not always consistent. For example, Liu *et al.* [LCT04] successfully explored all the 2D lattices that occur in 2D symmetry groups. Other works such [LLH04; LL07] show the power of lattice by expanding them to non-linear cases that models the geometric deformation that occur in real-world 3D objects. Some works [Pau+08a] simply do not talk about lattices at all, and treat translation

symmetry as a group consisting of generators just like any other symmetry. While admittedly, there is nothing wrong with this viewpoint, it makes it hard to reconcile the variety of works done in this field into a uniform nomenclature and mathematical framework.

In this thesis, I adhere to a classification scheme for symmetry that is shown in Figure 2.6. The generative models that we provide in the next chapter will also adhere to this classification. In fact, I show that this kind of classification generates 4 distinct generative models for all symmetries that can occur in 1D, 2D and 3D (as far as I am aware). Furthermore such a classification allows us to fully exploit the lattice framework we introduced in the previous section, which defines one the most important component of this thesis: the **asymmetric unit**

Symmetries in 1D

All symmetry groups of 1D configurations are subgroups of $\mathsf{E}(1)$

$$y = \mathsf{R}x + \mathsf{t} \tag{2.10}$$

where $\mathsf{t} \in \mathbb{R}$ is the 1D displacement of the point $\mathsf{x} \in \mathbb{R}$. Euclidean group in 1D, do not have rotations but can have reflections, therefore $\mathsf{R} = \pm 1$. Space groups in 1D represent all symmetry groups in 1D. Allowable operations in 1D space include translations parallel to the line and reflection perpendicular to the line.

Point Groups in 1D In a single dimension, the point group contains only the trivial group C_1 and the reflection group D_1. C_1 is the trivial group because it only contains the identity element while D_1 contain two elements: (1) the identity and (2) reflection about the perpendicular line passing thorough the invariant point.

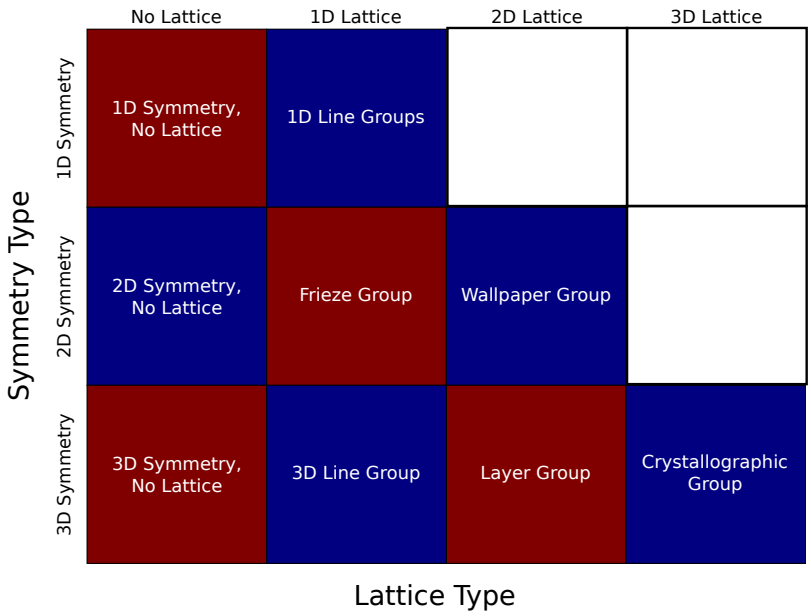


Figure 2.6: **Symmetry Groups**: We exhaustively describe the symmetry groups for the case of lattice and non-lattice symmetries in 1D, 2D and 3D. Some of them fall into categories that are described in popular literature, others are more arcane. Such a kind of classification allow us to also understand subgroup properties as the group elements G and the generators g increase from left to right.

1D Symmetry, No Lattice

Any of the space groups could present itself as a 1D or 2D lattice with no symmetry. Figure 2.7 shows the example of 1D symmetry with no lattice.

1D Line Groups [1D symmetry,1D Lattice]

One-dimensional line groups are a discrete group that describe the symmetries that occur along a line in the form of a repeating pattern. Although we briefly talked about one dimensional symmetries when we addressed them in the context about point group (C_1

Figure 2.7: **1D Symmetry with No Lattice**: An Example of an 1D infinite line that has no lattice but has a 1D symmetry. The line however has bilateral symmetry and it belongs to the group D_1. Similarly C_1 is also an example of a 1D symmetry with no lattice which contains the trivial transformation (the identity).

and D_1), line groups offer two more symmetries in one dimension : $p1$ and $p1m$ which can be visualized using a discrete set of points that occur along a line.

Symmetries in 2D

Similar to 1D, all symmetry groups of 2D configurations are subgroups of $\mathsf{E}(2)$. Euclidean transformations in 2D are represented by

$$y = \mathsf{T}x \tag{2.11}$$

where $\mathsf{T} \in \mathbb{SE}(2)$ which is isomorphic to the set of 3×3 matrices where $T = \begin{bmatrix} \mathsf{R} & \mathsf{t} \\ 0 & 1 \end{bmatrix}$ where R is an element of the Orthogonal group $\mathsf{O}(2)$ and $\mathsf{t} \in \mathbb{R}^2$ is the translational vector in 2D. The space groups in 2D are the symmetry groups of a 2D plane. Of special interest is the 2D point group:

Point Groups in 2D

Point groups in 2D can be classified into two specific families (1) the family of cyclic groups C_n and (2) dihedral groups D_n. Rotations in 2D are with respect to a single point

and reflections are with respect to a line contained in the 2D plane. Figure 2.9 and 2.8 provide an example of the Cyclic and Dihedral group respectively.

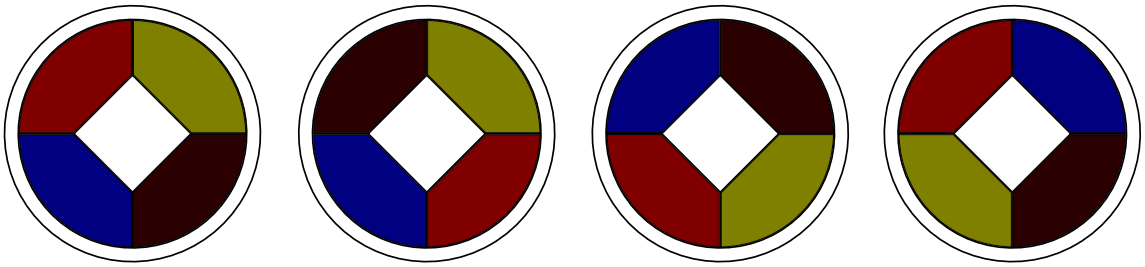


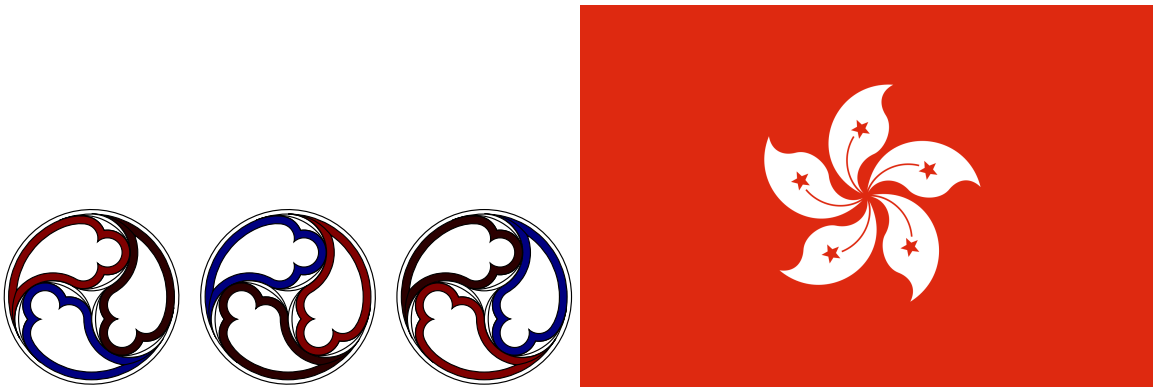
Figure 2.8: **Dihedral Group** (D_4) **in 2D**: Example of a window in the similar architectural period that exhibits Dihedral symmetry D_4. In addition to C_4 (not shown here) this window exhibits 4 more symmetries that arise as a consequence of reflection coupled with rotation. It is to be noted that none of these configurations can be obtained by pure rotations alone.

2D Symmetry, No Lattice

Any of the space groups could present itself in a 1D or 2D lattice setting. Among trivial example, C_1 is an example of a 1D symmetry with no lattice which contains the trivial transformation (the identity). In fact all the cyclic and dihedral groups do not have a lattice structure in 2D. Figure 2.9 shows an example of a 2D symmetry with no lattice.

Frieze Groups [2D symmetry,1D Lattice]

The two dimensional line groups also called the **Frieze groups** represent symmetries that keep a a line in 2D fixed. The symmetry operation (12) that are allowed in this group are (1) translations parallel to the line and (2) reflections parallel (3) reflections perpendicular to the line (4) glide reflections parallel to the line and (5) 180^0 rotations. Considering the



(a) Gothic window (b) Flag of Hong Kong

Figure 2.9: **Cyclic Group** (C_3) **in 2D**: A Gothic window [link] common in the 15th century that exhibits a cyclic symmetry C_3 defined in the Schönflies notation. This is a specific example of Axial symmetry that only has cyclic symmetry and no dihedral/ Improper rotations group. Every rotations of 120° about the center leaves the object unchanged (the color is given here to visually distinguish the three cases).

various combinations of the 4 symmetry operations, one might guess there are 26 groups $({}^4C_1 + {}^4C_3 + {}^4C_2 + {}^4C_1)$ that can be combined with translations to generate a valid symmetry group , but in reality, not all combinations of symmetry operations lead to a valid symmetry group There are 7 frieze groups shown in Figure 2.10 and they are commonly represented in the International Union of Crystallography (IUC) notation as $p1$, $p1m1$, $p11g$, $p11m$, $p2$, $p2mg$, $p2mm$. If we consider the lattices alone, Frieze groups can be seen as exhibiting one of two types of lattices (1) oblique ($p1$, $p2$) and (2) rectangular ($p11g$, $p11m$, $p2$, $p2mg$, $p2mm$). While the lattice is 1D, Frieze groups are a two dimensional symmetry group and the points in the unit cell form a 2D object as shown in Figure 2.10. Frieze groups are therefore an example of a sub-dimensional lattice that we introduced in Section 2.2 with the parameters of the lattice spanning the following space: A point in the lattice $\mathsf{u}'' \in \mathbb{L}_{2,1} \middle| \mathbb{L}_{2,1} = (\mathbb{SE}(2), ([0,1], \mathbb{Z}^1))$ with the perpendicular pro-

jection of the point from the translational direction represented by $\mathsf{v}'' \in \mathbb{R}$. The point in the asymmetric unit is specified by $0 \leq \mathsf{u} \leq a$ and the point in the unit cell $0 \leq \mathsf{u}' \leq 1$. The lattice is represented by a basis $\mathsf{B} \in \mathbb{R}$ the transformation of the lattice with respect to a coordinate frame of reference is given by $\mathsf{R} \in \mathbb{SO}(2)$ and $\mathsf{t} \in \mathbb{R}^2$. Figure 2.10 shows all the seven Frieze groups if they occur as window repetitions. I also show the meaning of the asymmetric units, unit cells and they really appear if they are taken out of the lattice and visualized.

Wallpaper Group [2D symmetry, 2D Lattice]

Wallpaper groups are the family of symmetry groups of a 2D symmetric object that exhibit a 2D lattice structure. There are 17 wallpaper groups that are obtained by a combination of 4 fundamental operations: translations, rotations, reflections and **glide reflections** (they are reflections where the planes of reflection are perpendicular to the 2D plane that contains the lattice). The 17 wallpaper groups are $p1$, $p2$ pm, pg, cm, pmm, pmg, pgg, cmm, $p4$, $p4m$, $p4g$, $p3$, $p3m1$, $p31m$, $p6$, $p6m$. All these group elements are explained in detail in the works of Liu *et al.* [LCT04]. Wallpaper groups can have one of 5 Bravais lattices: hexagonal($p3$, $p3m1$, $p31m$, $p6$, $p6m$), square($p4$, $p4m$, $p4g$), rectangular(pm, pg, cm, pmm,cmm), rhombic(pmg, pgg) or oblique($p1$, $p2$). Unlike Frieze groups, the wallpaper groups are not a sub-dimensional lattice with the parameters of the lattice spanning the following space: A point in the lattice $\mathsf{u}'' \in \mathbb{L}_{2,2} \Big| \mathbb{L}_{2,2} = \left(\mathbb{SE}(2), \left([0,1]^2, \mathbb{Z}^2\right)\right)$ with the perpendicular projection of the point from the translational direction represented by $\mathsf{v}'' = \emptyset$. The point in the asymmetric unit is specified by $\mathsf{0}^2 \leq \mathsf{u} \leq [a, b]$ and the point in the

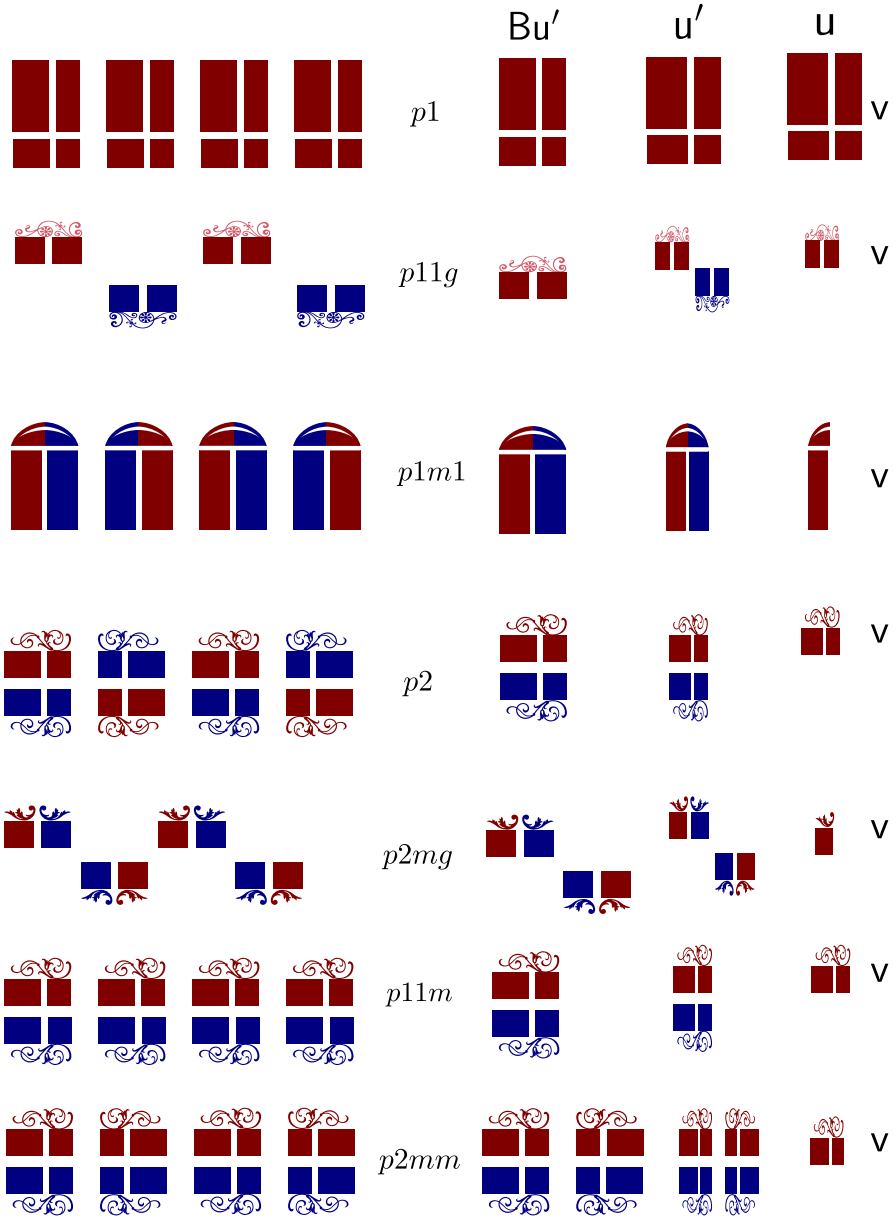


Figure 2.10: **Frieze symmetry in window architecture and asymmetric units**(a.k.a Motifs [LCT04]): Images that I generate to illustrate 7 Frieze groups along with the points in the asymmetric units u, v and unit cells $\mathsf{u}\prime$ for common window styles that occur in urban environments. Blue is used to represent symmetries that are obtain by reflection maroon is used to represent the unreflected symmetries. A point in the lattice $\mathsf{u}\prime\prime \in \mathbb{L}_{2,1} \middle| \mathbb{L}_{2,1} = (\mathbb{SE}(2), ([0,1], \mathbb{Z}^1))$ with the perpendicular projection of the point from the translational direction represented by $\mathsf{v}\prime\prime \in \mathbb{R}$. The point in the asymmetric unit is specified by $0 \leq \mathsf{u} \leq a$ and the point in the unit cell $0 \leq \mathsf{u}\prime \leq 1$. The lattice is represented by a basis $\mathsf{B} \in \mathbb{R}$.

(a) Frieze Group in 2D (b) 3D Frieze Group

Figure 2.11: **2D symmetries with Lattices**: Real world example of 2D symmetries with Lattice. We present here a case of (a) Frieze group (every row in the pattern belongs to a different Frieze Group) and (b) wallpaper group. This is an unorthodox example of wallpaper group $p3m1$.

unit cell $0^2 \leq \mathsf{u}\prime \leq 1^2$. The lattice is represented by a basis $\mathsf{B} \in \mathbb{R}^{2\times 2}$ the transformation of the lattice with respect to a coordinate frame of reference is given by $\mathsf{R} \in \mathbb{SO}(2)$ and $\mathsf{t} \in \mathbb{R}^2$.

Symmetries in 3D

All symmetry groups of 3D configurations are subgroups of $\mathsf{E}(3)$, the Euclidean group in 3D. The Euclidean group is isomorphic to the set of 4×4 matrices representing a Euclidean

transformation by:

$$y = \mathsf{T}\mathsf{x} \tag{2.12}$$

where the matrix $\mathsf{T} \in \mathbb{R}^{4\times4}$ has structure of the form $\mathsf{T} = \begin{bmatrix} \mathsf{R} & \mathsf{t} \\ 0 & 1 \end{bmatrix}$ where R is an element of the Orthogonal group $\mathsf{O}(3)$ and $\mathsf{t} \in \mathbb{R}^3$ is a translation vector. The Euclidean group consists of three fundamental operations that make up the matrix T: (1) translation (2) rotation and (3) reflection. **The Space groups** are a subgroup of $E(3)$ that represent the symmetry group of an object $\mathcal{X} = \{\mathsf{x}_1, \mathsf{x}_2 \ldots\}\, \forall \mathsf{x} \in \mathbb{R}^3$ in 3D. Since Euclidean transformations include translations ,rotations and reflections, it constrains $\mathsf{R} \in \mathbb{SO}(3)$ to be an orthogonal matrix $|\mathsf{R}| = \pm 1$. Geometrically, an orthogonal matrix represents a rotation or a reflection or a combination of the two. Space groups have a unique family of subgroups that require special attention, the **point groups**

Point Group

In $\mathbb{R}^3$, a point group is a subgroup of $\mathsf{O}(3)$. A space group is a point group if the action of its element leaves at least one point fixed in location [McW02]. This implies that the translation vector t is Eq. 2.12 has to be a zero vector. Within the set of 3D Euclidean transformations $\mathsf{E}(3)$, the set of rotations and reflections fully comprise the point group. The family of point groups can be one of three possible groups (1) Axial (2) Platonic or (3) the continuous group, $\mathsf{O}(3)$.

(1) Axial Groups: Axial groups are a subset of the family of point groups that leave all the points along an axis unchanged. They have a single axis called the principal axis

(a) **Dihedral (D_{12}) Symmetry in 3D** (b) Dihedral Symmetry (3D) with 1D Lattice

Figure 2.12: **Axial Groups in 3D**: (a) A Gothic window common in the 15th century that exhibits a Dihedral symmetry D_{12} defined in the Schönflies notation. This is a real 3D example as opposed to the 2D orthographic image generated in Figure 2.9. Considering the image of the window as a 2D Cyclic group will not be beneficial because of the subtle gradations in depth of the window and the projection of the camera relative to it. (b) Another example of an Axial group this time, an apartment complex in Matveyevskoye district, Moscow exhibiting a 1D lattice in 3D. This is discussed in Section 2.3.

that has a higher order than any other axis [McW02]. There are seven families of axial groups that can be represented in the so-called Schönflies notation: $C_n, S_{2n}, C_{nh}, C_{nv}, D_n$, D_{nd}, D_{nh}. Symmetry elements are denoted by i for centers of inversion, C for proper rotation axes, σ for mirror planes, and S for improper rotation axes (rotation-reflection axes). C and S are usually followed by a subscript number (denoted n) denoting the order of rotation possible. While these represent the discrete groups, as $n \to \infty$, these groups include the continuous group $\mathsf{O}(3)$. Figure 2.12 shows an example of 3D Axial Group that is distinguished from the 2D Axial group discussed in Figure 2.9.

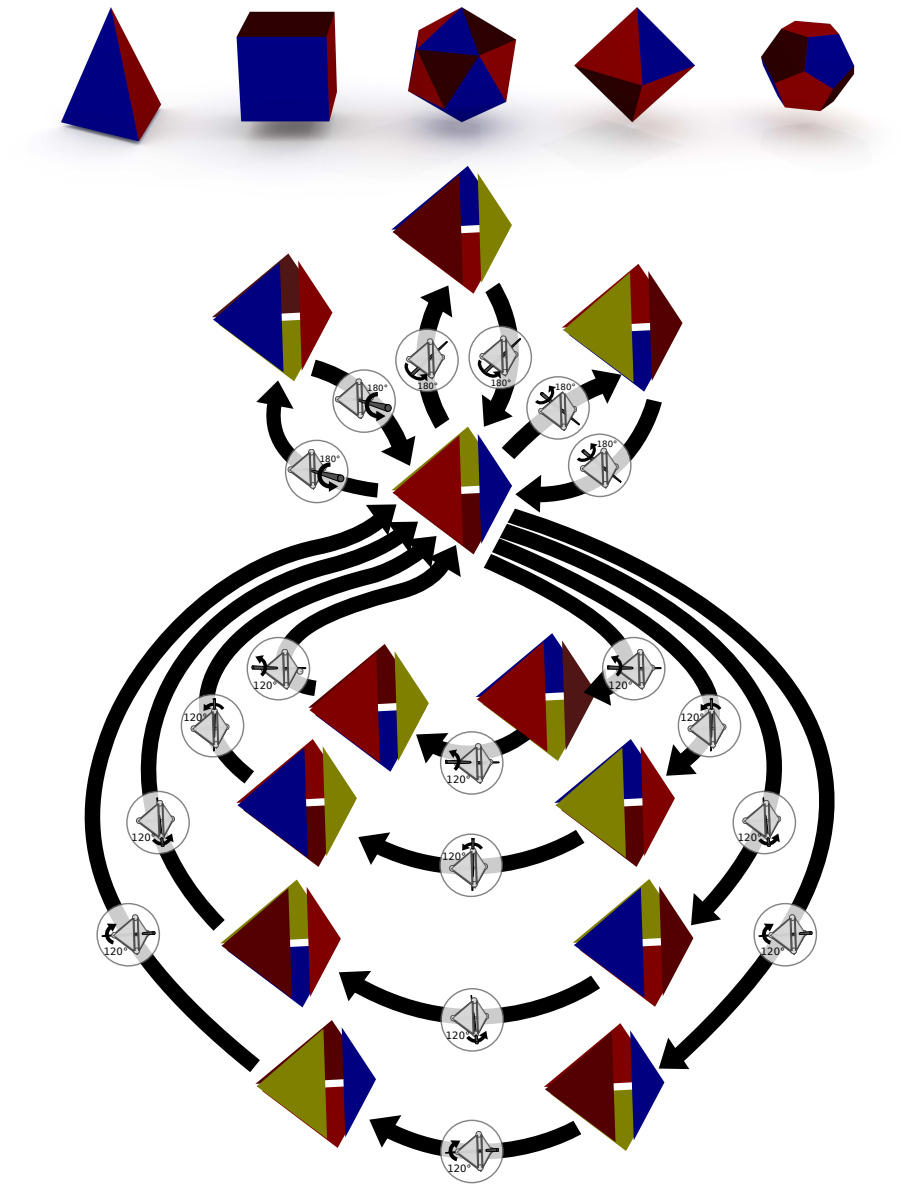


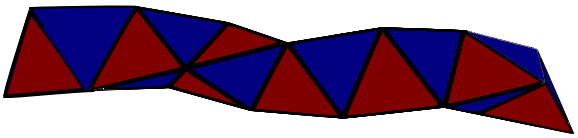
Figure 2.13: **Platonic Solids Group**: The top row shows the 5 platonic solids (Tetrahedron, Cube, Icosahedron, Dodecahedron and Octahedron from left to right). The symmetry groups of these solids are the platonic solids group given by T, T_d, T_h, O, O_h, I, I_h types of symmetry groups. As an example, we explore one of the groups T which the Chiral Tetrahedral group which describe the symmetries of a tetrahedron. This group elements contain transformations that correspond to rotations along three edges and the axes of the tetrahedron. This produces 12 distinct configurations depicted in the above figure.

(2) Platonic Groups: The symmetry groups of the Platonic solids are a special class of three-dimensional point groups known as Platonic groups. The high degree of symmetry of the Platonic solids can be interpreted in a number of ways. Most importantly, the vertices of each solid are all equivalent under the action of the symmetry group, as are the edges and faces. The action of Platonic groups on the relevant platonic solid leaves its center unchanged and it is therefore a point group. The seven **Platonic groups** represented in the Schönflies notation are T, T_d, T_h, O, O_h, I, I_h. T (The chiral tetrahedral group T) has the rotation axes of a tetrahedron (three $2-$fold axes and four $3-$fold axes). O (the chiral octahedral group O) has the rotation axes of an octahedron or cube (three $4-$fold axes, four $3-$fold axes, and six diagonal $2-$fold axes). I (the chiral icosahedral group) indicates that the group has the rotation axes of an icosahedron or dodecahedron (six $5-$fold axes, ten $3-$fold axes, and 15, $2-$fold axes). Figure 2.13 shows the different configurations of the Chiral Tetrahedral group T.

3D Symmetry, No Lattice

Not all 3D symmetries have a lattice structure. All the symmetry operations described in Definition 12 (except translation) that occur in isolation have a non-lattice symmetry.For example, Figure 2.14b shows an example of a 3D symmetry that has no lattice structure for the screw operation defined in Definition 12. This example is particularly counter intuitive, since all screw operations can be easily seen as having a lattice (see Figure 2.15, for a screw operation with a lattice). Sometimes symmetry operations can even occur in combination to still form a non-lattice symmetry group. In Figure 2.14a, we see an example of multiple

(a) Dihedral group D_N



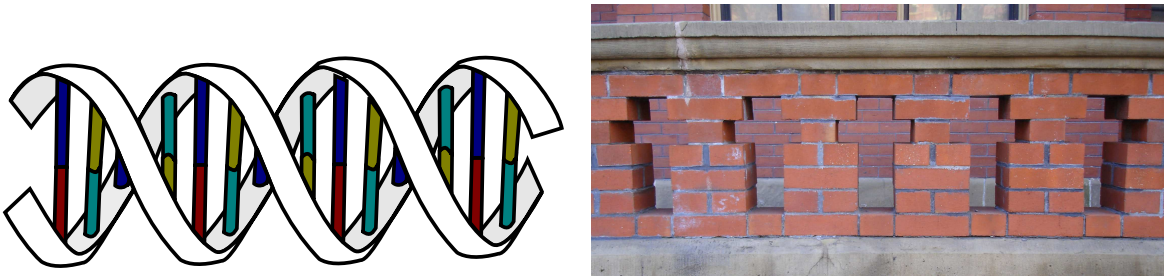
(b) Irrational Helix

Figure 2.14: [**3D symmetry, No lattice**]: (a)Consider the case of the Royal albert Hall, a circular building in London. This exhibits two of the symmetry operations defined in Definition 12 (reflection and rotation). They combine to form the Dihedral group D_N, an Axial group in 3D that does not exhibit any lattice structure. (b) Another example of an 3D symmetry that has **no lattice** is the irrational helix. An irrational helix is a linear stacking of regular tetrahedra, arranged so that the edges of the complex that belong to a single tetrahedron form three intertwined helices. The repetition is along the helical curve and there is no repetition as per the definition of lattice defined in Eq. 2 along the axis of the helix. This is an example corresponding to the **screw operation** defined in which occurs without any lattice. This is one of the more difficult examples to visualize.

symmetry operations that occur together to form a complex symmetry group the Dihedral group D_N, similar to the Gothic window shown in Figure 2.12a (which also exhibits a 3D symmetry with no lattice)

3D Line Groups [3D symmetry, 1D Lattice]

An element of a 3 dimensional line group leaves a line in 3D invariant. Intuitively, there are three operations that leave a infinite line invariant (1) translations parallel to the line (2) rotations about the line and (3) reflection about any plane passing through the line. In comparison with 2D line groups, 3D line groups have 3DOF which allow any/all of the



(a) Double Helix (b) 3D Frieze Group

Figure 2.15: **3D symmetry with 1D Lattice**: (a) The double helix is an example of a 3D symmetry with a 1D lattice. The unit cell is equal to one pitch of the helix and the asymmetric unit is one half of the unit cell.A point in the lattice $\mathsf{u}'' \in \mathbb{L}_{3,1} \middle| \mathbb{L}_{3,1} = \left(\mathbb{SE}\left(3\right), \left(\left[0,1\right]^1, \mathbb{Z}^1\right)\right)$ with the perpendicular projection of the point from the translational direction represented by $\mathsf{v}'' = \mathbb{R}^2$. The point in the asymmetric unit is specified by $0 \leq \mathsf{u} \leq a$ and the point in the unit cell $0 \leq \mathsf{u}' \leq 1$. The lattice is represented by a basis $\mathsf{B} \in \mathbb{R}$ the transformation of the lattice with respect to a coordinate frame of reference is given by $\mathsf{R} \in \mathbb{SO}\left(3\right)$ and $\mathsf{t} \in \mathbb{R}^3$. (b) A 3D Frieze group, which is, in reality, a Layer group $P1$ with a monoclinic lattice according to our classification scheme.

above actions. Rotations about the line in combination with translations parallel to the line are of particular interest as this is common in architecture (*eg.* cylindrical buildings). As a result, these groups can also be interpreted as patterns of 2D symmetry wrapped around a cylinder n times and infinitely repeating along the cylinder's axis, much like the three-dimensional point groups and the Frieze groups. The double helix shown in Figure 2.15 is another important example of a 3D line group with translation and rotations occurring together. 3D line groups exhibit a lattice structure parallel to the line and they are isomorphic to the translational symmetry group parallel to the line.

Figure 2.16: **3D symmetry 2D lattice**: Richard Seifert's Centre Point, a building in Central London has a set of windows that exhibit a 3D symmetry in a 2D lattice structure. Furthermore, the windows have a bilateral symmetry, the series of windows belong to the Layer group Pm. The unit cell extends from the center of one miller to its adjacent pillars that occur between the windows and asymmetric unit is half of the unit cell. A point in the lattice $\mathsf{u}'' \in \mathbb{L}_{3,2} \middle| \mathbb{L}_{3,2} = \left(\mathbb{SE}\left(3\right), \left(\left[0,1\right]^2, \mathbb{Z}^2 \right) \right)$ with the perpendicular projection of the point from the translational direction represented by $\mathsf{v}'' = \mathbb{R}^1$. The point in the asymmetric unit is specified by $0^2 \leq \mathsf{u} \leq [a,b]$ and the point in the unit cell $0^2 \leq \mathsf{u}' \leq 1^2$. The lattice is represented by a basis $\mathsf{B} \in \mathbb{R}^{2\times 2}$ the transformation of the lattice with respect to a coordinate frame of reference is given by $\mathsf{R} \in \mathbb{SO}\left(3\right)$ and $\mathsf{t} \in \mathbb{R}^3$.

Layer Group [3D symmetry, 2D Lattice]

Layer groups capture those 2D symmetries that occur in 3D. Layer groups are by far the most common architectural symmetry groups that occur in urban scenes, due to periodic window patterns (3D) objects that appear on the face in 2D lattice. There are 80 families of layer groups that can be classified into 6 categories based on the 6 types of 2D Lattices - *triclinic, monoclinic, orthorhombic, tetragonal, trigonal and hexagonal*. Similar to 3D line groups that allow reflection and rotation about the 3D line, layer groups allow rotations about the lattice center and also reflections along a plane that is parallel to the plane of the 2D lattice. Every lattice center of a Layer group forms an Axial group. Figure 2.16 shows a real world example of a ***monoclinic, orthogonal Layer*** group represented by $p211$ (where p stands for primitive cell and 2 represent the highest order of symmetry (this case, 2) and the 11 represent the translational symmetry in the two perpendicular directions.

Crystallographic Groups [3D symmetry, 3D Lattice]

Crystallographic groups capture 3D symmetries that occur in 3D lattices. The family of crystallographic group consists of 230 groups where every group is some combination of one of the 32 families crystallographic point group with any of the 14 Bravais lattices (disallowing some combinations). The 32 crystallographic point groups are one of 7 possible groups: $C_n, C_{nv}, C_{nh}, D_n, D_{nh}, S_{2n}$ with only allowable parameters for n as $1, 2, 3, 4$ and 6 according to the crystallographic restriction theorem [BCK03](3 of those configurations not allowed, which brings the total to $7 \times 5 - 3 = 32$). The 14 Bravais lattices can be

further categorized into 6 categories as in the 2D case with a variety of centering types for the cells of the lattices like Centered, Base-centered, Body-centered and Face-centered. Once again, not all combination of lattice systems are allowed to occur with all centering types and 14 unique lattices have been identified. Both layer groups and 3D line groups are a special case of the crystallographic group.

Conclusion

I conclude this chapter by summarizing how I have substantiated the first claim of my thesis

> *I use the mathematical theory of symmetry from authoritative texts to synthesize an overview of symmetry that is immediately beneficial to the computer vision community.*

I have provided a formal definition of symmetry, lattice and asymmetric unit and several related and important concepts, obtained from a variety of authoritative texts on symmetry. I have demonstrated how we can use these definitions to categorize the symmetries that we observe in nature and provided many examples from a variety of scenarios that fall within our classification scheme. While the primary focus of this thesis is urban scene understanding, I have provided examples that fall outside this realm to further show that our classification scheme can be applied to many symmetries beyond the ones that occur in architecture.

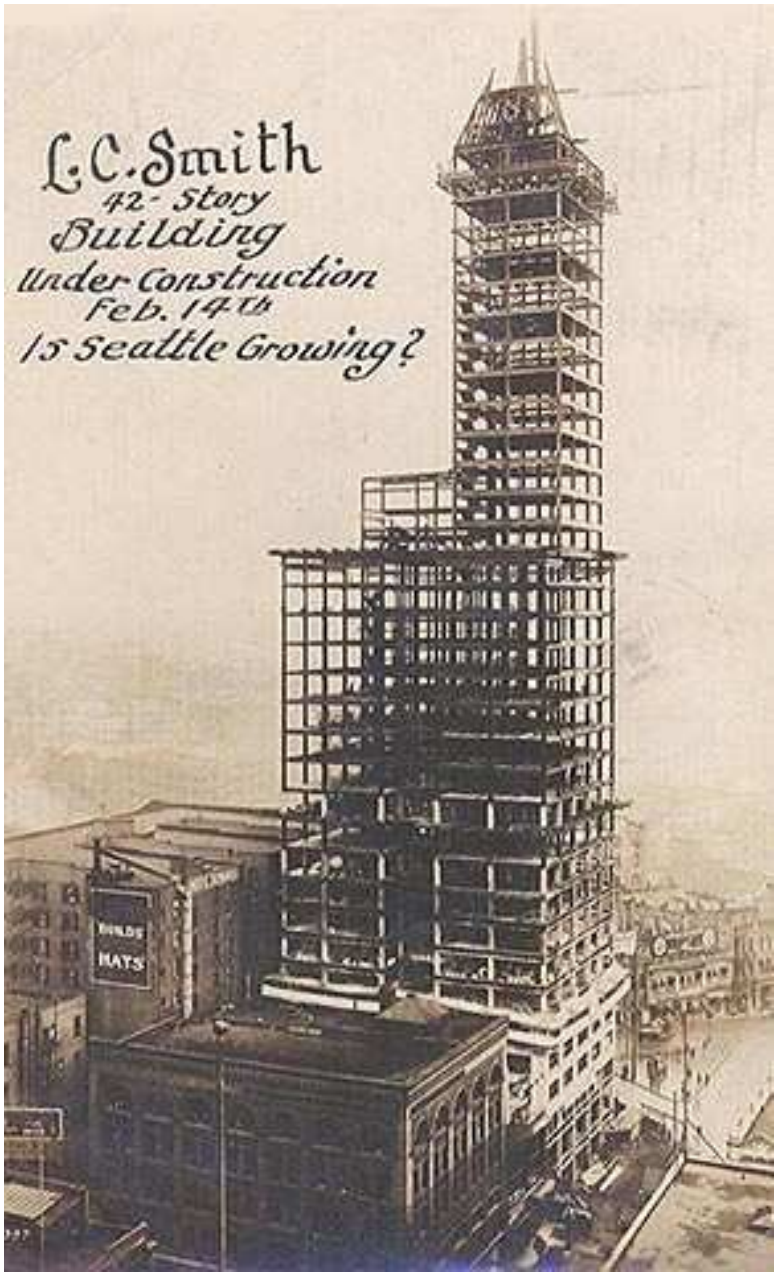


Figure 2.17: **3D symmetry 3D lattice**: Chicago's Home Insurance Building under construction in 1885. Scaffoldings are typically 3D lattices and they can be inferred from images since we can "see through" the building. In this case, we can assume that the lattice belongs to the crystallographic group $P1$.

Chapter 3

A GENERATIVE MODEL OF SYMMETRY

Introduction

In this chapter, I aim to substantiate the second claim of my thesis:

> *I provide generative models for synthesizing symmetries in 1D, 2D and 3D that encompass a larger variety of symmetries than the state of the art.*

In particular, I develop a mathematical framework for synthesizing symmetries in 1D, 2D and 3D that is immediately beneficial to the computer vision community. In the previous chapter, we defined the many related terminologies that make up symmetry and lattices and demonstrated how we can apply these definition to categorize the symmetries observed in urban scenes. We also synthesized a mathematical framework for describing these entities from texts on symmetry such as [CS13; Wey15; Mar12; McW02; Mül13]. In this chapter we will see how these mathematical framework combine in the different blocks of our classification scheme on symmetry providing us with a probabilistic model for generating symmetry for each category.

A probabilistic model is useful for predicting an observation. It is composed of relationships and variables, where variables are some parameters of interest that can be quantified and the relationships are comprised of mathematical operators. There are several types of mathematical models used in nearly all field of science of engineering. Better models predict the observations more accurately and often supersede an older model which makes a poorer prediction. While accuracy in prediction is certainly desirable, the quantifiable metrics that make one model better than the other is application dependent and choosing the right model is an engineering choice that is influenced by practical issues. In this thesis, I provide a model of symmetry that is useful in the domain of computer vision and computer graphics where the observation is typically in the form of image(s) observing a 3D scene or 3D points directly on the structure.

A generative model is a probabilistic model that can generate not only the observation given a set of variables, but also generate the variables (from a "prior" distribution) itself thereby providing both the input and the output. In contrast, discriminative models take an observation as an input and provide a model for predicting the variables that generate the observation. Since generative models generate both the input and output of the system, they are particularly useful for incorporating domain knowledge on any of the variables of the model (if it exists). The domain knowledge is typically in the form of a probability density function and sampling of this function generates an instance of variable that the function models.

A Bayes net is a useful graphical representation [Pea88] of a generative model. It is a directed acyclic graph whose nodes represent the set of variables and edges represent conditional probability densities of the variables that it connects. It also models the causal

relationship of the variables (hence, "directed") with the parent node in the graph affecting its child node(s). A arrow is used to point from the parent to the child node to graphically depict the parent-child relationship. Probabilistically, the conditional density between the parent node and its child is a probability function with the probability of certain value of the parent node an input and the probability of a value of the child node as the output. A Bayes net models the joint probability density of a set of variables that constitute it, with the unique connections between the variables, providing a high level means to factorize the joint distribution into a product of conditional densities between parent and child nodes. Mathematically, this is defined in [DK17] as

$$p(\Theta) \triangleq \prod_j p\left(\theta_j \middle| \pi_j\right) \tag{3.1}$$

where $\Theta = \{\theta_1 \dots \theta_n\}$ are the (random) variables of the model $p(\Theta)$ is the joint density, $p\left(\theta_j \middle| \pi_j\right)$ is the density of the child θ_j provided the parents take on the value π_j .

In this thesis, we typically use the multivariate Gaussian distribution with probability density

$$\mathcal{N}(\theta, \mu, \Sigma) = \frac{1}{\sqrt{2\pi\Sigma}} \exp\left(-\frac{1}{2} \|\theta - \mu\|_\Sigma^2\right) \tag{3.2}$$

where $\mu \in \mathbb{R}^n$ is the mean and $\Sigma \in \mathbb{R}^{n \times n}$ is the covariance matrix, and

$$\|\theta - \mu\|_\Sigma^2 \triangleq (\theta - \mu)^T \Sigma^{-1} (\theta - \mu) \tag{3.3}$$

is the squared Mahalanobis distance. Under this assumption, we can represent the probabilistic generative model given in Equation 3.1 as a measurement prediction function that

is corrupted by Gaussian noise given by

$$\mathsf{z} = h\left(\theta_j, \pi_j\right) + \eta$$

where $h\left(\cdot\right)$ is the measurement function and η is the measurement noise given by $\mathcal{N}\left(\mathsf{z}, 0, \Sigma_{\mathsf{z}}\right)$ is the noise on the measurement z that is a zero-mean Gaussian having a covariance given by Σ_{z}.

I validate my generative model by providing an **optimization framework** for each specific instance of symmetry and lattice combination. The optimization framework utilizes the generative model to infer the unknown variables in Θ from a set of observation of noisy points of the structure. We graphically represent the optimization as a **factor graph,** a bipartite graph that represents the factorization of a function; in this case the energy function that corresponds to the optimization problem. Factor graphs has been successfully used for modeling and solving of large-scale inference problems in robotics [DK17] and we show this approach can be effectively applied in estimating the parameters of symmetry. In Chapters 5 and 6 we show how this factor graph can be modified to address the case when these symmetric points are observed in an image, by modeling the projection function into the above factors.

This chapter comprises of 3 sections corresponding to symmetries in 1D 2D and 3D. Each section discusses the case of different dimensions of lattices that could occur for that particular symmetry configuration. This matches with classification for symmetry that we provided in Chapter 2. A user can use our generative model by first recognizing the category of symmetry that he/she wants to generate and use the generative model that

corresponds to that category of symmetry.

Generating Symmetries in 1D

No Lattice

In this section, I develop the generative model for generating 1D points that are symmetric and do not exhibit any lattice structure. Figure 3.1(a) is an example of this configuration.

The generative model that corresponds to this specific instance of symmetry is shown in Figure 3.1(b). The model comprises of a single child node $\mathsf{x} \in \mathcal{X}$ that correspond to a point in the symmetric structure. Every point x has a corresponding Wyckoff position inside the structure which indicates if the point belongs inside the asymmetric unit or if it is reflected from its corresponding location in the asymmetric unit. This is indicated by the variable $\mathsf{w} \in \{\{a, b\}, \{0, 1\}\}$. The international table of crystallography uses letters from $\{a \ldots z\}$ to indicate the multiplicity of the point. The lowest alphabet corresponds to the point with the lowest multiplicity. If there is a tie, the user has to refer to the crystallographic table to determine the exact location of the current point. For this specific instance of symmetry, w is a tuple where the first element is a binary variable that can take on one of two possible combinations (a, b) and the second element is also a binary variable that can take on one of the two possible values $(0, 1)$. Therefore there are 4 legal combinations for this variable $(a, 0), (a, 1), (b, 0), (b, 1)$. In addition to this, every point x has a corresponding location inside the asymmetric unit. This location is described by the variable $\mathsf{v} \in [0, \infty$. This is

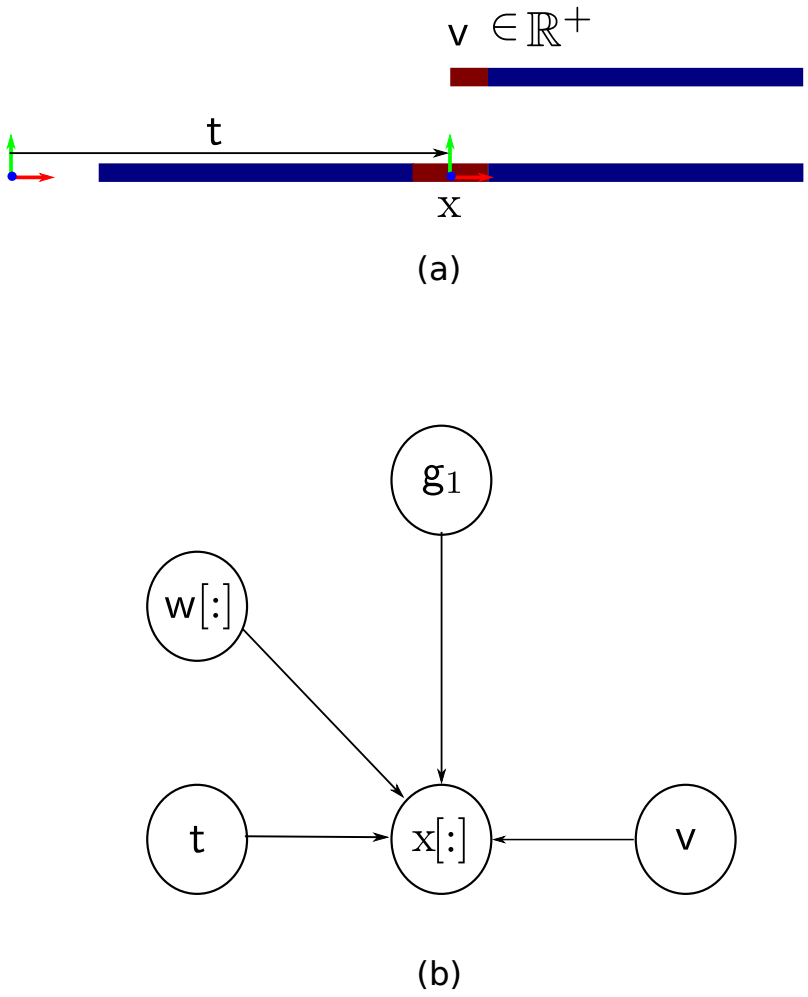


Figure 3.1: **Generative model for synthesizing 1D symmetry with no lattice**. (a) I show an infinite line that is bilaterally symmetric about its origin. The line itself is translated by a translation vector t. Since the line is bilaterally symmetric, we can generate the entire line by using only half the line and this forms the asymmetric unit. Since there is no lattice $\mathsf{u} = \emptyset, \mathsf{v} \in \mathbb{R}^+$ while points in the 3D structure is given by $\mathsf{x} \in \mathbb{R}$. (b) The generative model depicts a probabilistic model for generating a point x provided, we are given its corresponding point in the asymmetric unit v and its Wyckoff parameters given by $\mathsf{w} \in \{\{a, b\}, \{0, 1\}\}$ where the position $\mathsf{w}(0) = a$ corresponds to the venter of the line whose points have a multiplicity of 1 and any other location corresponds to the position $\mathsf{w}(0) = b$ where the points have a multiplicity of 2. If $w(1) = 0$, this corresponds to the point in the asymmetric unit and if $w(1) = 1$, then this corresponds to the reflected point from it corresponding point in the asymmetric unit. g_1 is the generator that corresponds to reflection about the vertical axis passing through the origin.

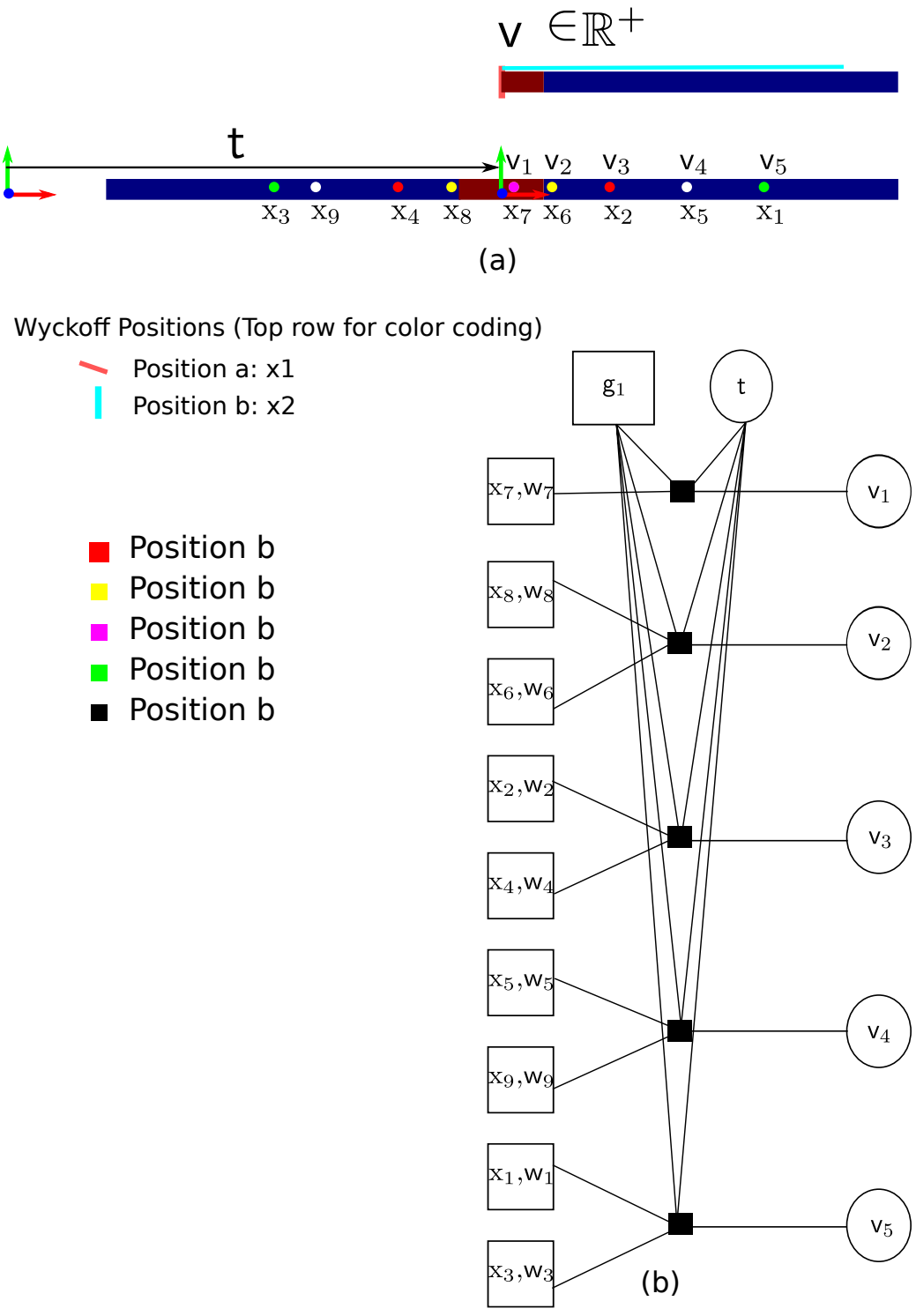


Figure 3.2: **Factor graph for the optimization problem defined in Section 3.1**. We are given a set of 9 measurements of point $\mathcal{X} \in \mathrm{x}_1 \ldots \mathrm{x}_9$ and its Wyckoff indicator variables $\mathcal{W} = \{w_1 \ldots w_9\}$. Our goal is to estimate the unknown location of points in the asymmetric unit $\mathcal{V} = \{\mathrm{v}_1 \ldots \mathrm{v}_5\}$ that generate these 9 points $\mathcal{X}$. The generator g_1 corresponds to reflection about the vertical axis passing through the origin and t is the translation vector.

because we choose the axis of reflection to pass through the origin leading to a convenient expression for the generator $\mathsf{g}_1 = [-1]$. In addition to this, we allow a translation of the entire symmetric structure from an arbitrarily assigned world coordinate frame we call this translation $\mathsf{t} \in \mathbb{R}$.

The joint probability distribution that this generative model describes is simply given by the single conditional distribution

$$p\left(\Theta\right) \triangleq p\left(\mathcal{X} \middle| \mathcal{V}, \mathsf{g}_1, \mathcal{W}, \mathsf{t}\right) \tag{3.4}$$

where $\mathcal{X} = \left\{\mathsf{x}_j\right\}_{j=1}^{n} \Big| \mathsf{x}_j \in \mathbb{R}$ is the set of points that are symmetric. $\mathcal{V} = \left\{\mathsf{v}_p\right\}_{p=1}^{P} \Big| \mathsf{v}_p \in [0, \infty$ are the set of points in the asymmetric unit, $\mathcal{W} = \left\{\mathsf{w}_p\right\}_{j=1}^{n} \Big| \mathsf{w} \in \left\{\left\{a, b\right\}, \left\{0, 1\right\}\right\}$ are the set of Wyckoff position for every point in $\mathcal{X}$ and $\mathsf{g}_1 = -1$ is the generator corresponding to reflection and finally $\mathsf{t} \in \mathbb{R}$ is the translation vector. There is a subtle modeling issue to be noted: we assume that the parent child relationship for every node here is known. This is modeled by the variable $p \in \mathsf{P}$ as a correspondence between a point in the asymmetric unit v_p and the symmetric point x_j which indicates the index between the corresponding variables as $\mathbf{p_j} \in \mathsf{P} \Big| \mathbf{p_j} = p$.

We can represent the generative model by means of a measurement function given by

$$\mathsf{x}_j = h_{10}\left(\mathsf{v}_p, \mathsf{g}_1, \mathsf{w}_j, \mathsf{t}\right) + \eta_{10} \tag{3.5}$$

where $\eta_{10} = \mathcal{N}\left(\mathsf{x}, 0, \sigma\right)$ is a Gaussian-corrupted noise on the point x on the symmetric

structure with a covariance σ. The measurement function $h_{10}\left(\mathsf{v}_p, \mathsf{g}_1, \mathsf{w}_j, \mathsf{t}\right)$ is given by

$$h_{10}\left(\mathsf{v}_p, \mathsf{g}_1, \mathsf{w}_j, \mathsf{t}\right) \equiv \begin{cases} \mathsf{g}_1^{\mathsf{w}_j(1)}\mathsf{v}_{p_j} + \mathsf{t} & \mathsf{w}\left(0\right) = b \\ \mathsf{t} & \mathsf{w}\left(0\right) = a \end{cases} \tag{3.6}$$

which describes the following process, to generate a point $\mathsf{x}_j \in \mathbb{R}$ in a non-lattice setting shown in Figure 3.1, we start from its corresponding point inside the asymmetric unit v_{p_j}. If the point is in a general position $\left(w\left(0\right) = b\right)$ (as all three points are in the figure) it has the highest multiplicity ($\times 2$) and can belong to either the unreflected region or the reflected region. However if the point is in the special position its multiplicity is 1 and this corresponds to the location of the line of reflection. Since for this example, this is chosen as the origin, $\mathsf{v}_{p_j} = 0$.

If we know the location of point in the general position only, Equation 3.6 reduces to

$$h_{10}\left(\mathsf{v}_p, \mathsf{g}_1, \mathsf{w}_j, \mathsf{t}\right) \equiv \mathsf{g}_1^{\mathsf{w}_j(1)}\mathsf{v}_{p_j} + \mathsf{t} \tag{3.7}$$

The likelihood function, a function of the parameters of the model $\mathcal{V}, \mathsf{t}, \mathcal{W}, \mathsf{g}_1, \mathsf{P}$ and the observed data x_j is given by

$$\mathcal{L}\left(\mathcal{X}; \mathcal{V}, \mathsf{t}, \mathcal{W}, \mathsf{g}_1, \mathsf{P}\right) \propto \prod_{j=1}^{n} \exp \left\| \frac{\left(\mathsf{x}_j - \mathsf{x}_j^{\dagger}\right)^2}{2\sigma} \right\| \tag{3.8}$$

where the $\mathsf{x}_j^{\dagger}$ is the predicted location of the point using the measurement prediction function given in Equation 3.5 x_j is the observed location.

The unknown variables in the above problem is the location of the point in the asymmetric

unit $\mathcal{V}$ and the translation of the symmetric structure t with respect to an arbitrarily assigned world co-ordinate frame. We assume that we know the value of the correspondence information P, the generator g_1 and the Wyckoff positions $\mathcal{W}$.

To determine the optimal values of these, we take the negative log likelihood which translates to a standard least squares problem, whose optimal value can be determined using non-linear optimization techniques such as Gauss Newton and Levenberg–Marquardt

$$-\log \mathcal{L}\left(\mathcal{X}; \mathcal{V}, \mathsf{t}, \mathcal{W}, \mathsf{g}_1, \mathsf{P}\right) = \frac{1}{2\sigma^2} \sum_{j=1}^{n} \left\| \mathsf{x}_j - \mathsf{x}_j^{\dagger} \right\|^2 \tag{3.9}$$

and the optimal value of the unknown variables given by $(\mathcal{V}^\star, \mathsf{t}^\star)$ is given by

$$(\mathcal{V}^\star, \mathsf{t}^\star) = \arg\max_{\mathcal{V}, \mathsf{t}} \left(-\log \mathcal{L}\left(\mathcal{X}; \mathcal{V}, \mathsf{t}, \mathcal{W}, \mathsf{g}_1, \mathsf{P}\right)\right) \tag{3.10}$$

Figure 3.2 shows a factor graph that corresponds to this optimization problem for a small example setup. All the points shown in this example are present in the general position and no points are in the special position. Therefore the location of all the points in the asymmetric unit is unknown. In the factor graph, circular nodes are used to show the variables that are being optimized and the rectangular nodes are used to depict observed variables. The correspondence information P is implicitly modeled by the link between the different nodes of the graph.

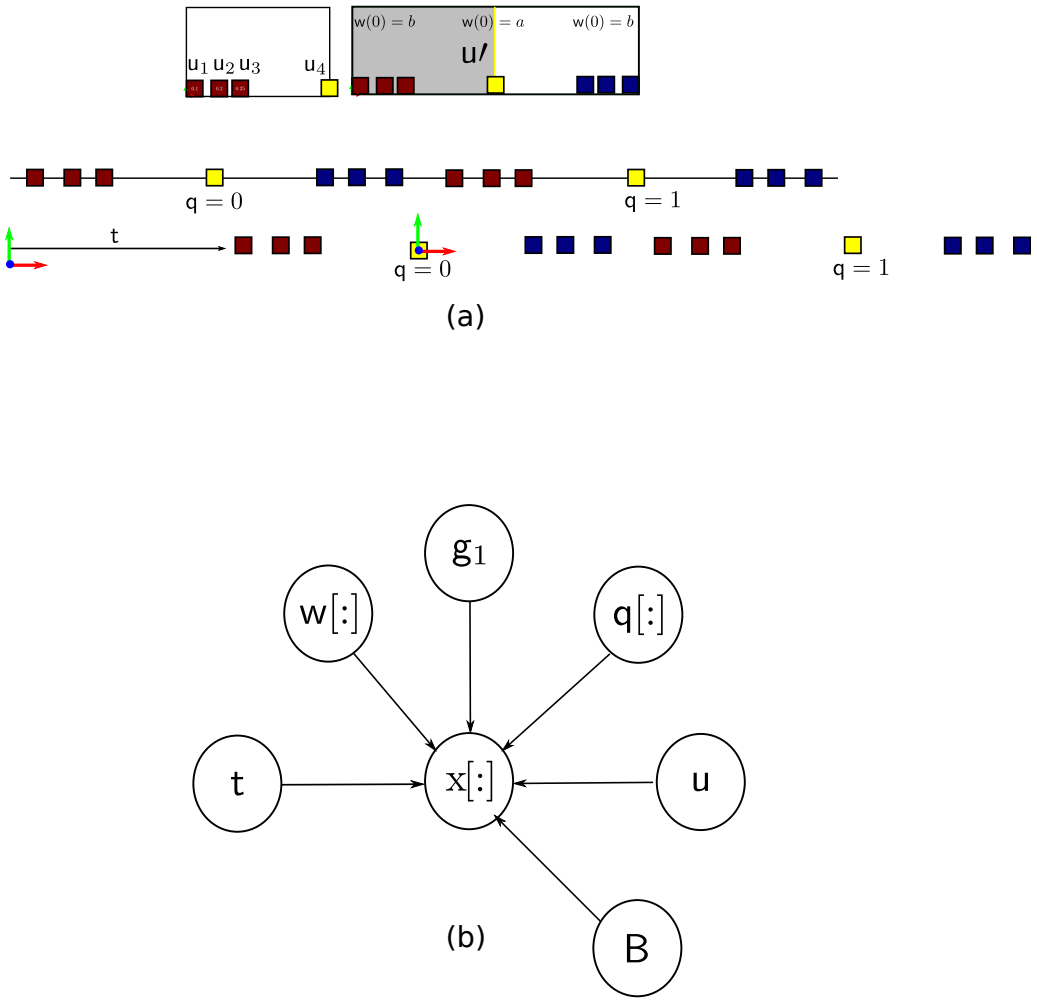


Figure 3.3: **Generative model for synthesizing 1D symmetry with 1D lattice**. (a) In the top row we show 4 points in the asymmetric unit $\mathcal{U} = \{\mathsf{u}_1, \mathsf{u}_2, \mathsf{u}_3, \mathsf{u}_4\}$. The origin of the asymmetric unit is chosen at 0 and the reflection axis is at 0.5. Therefore point u_4 is at a special position while the others are at the general position. $\mathsf{u}\prime$ represents the points in the unit cell. The points in the lattice $\mathsf{u}\prime\prime$ are the points in the lattice and finally the observed points x which is a translated version of the points in the lattice by a translation vector t. (b) In addition to the variables already discussed for the case of no lattice, the generative model for this case consists of two additional variables, the basis of the lattice $\mathsf{B} \in \mathbb{R}$ and the Miller indices $\mathsf{q} \in \mathbb{Z}$.

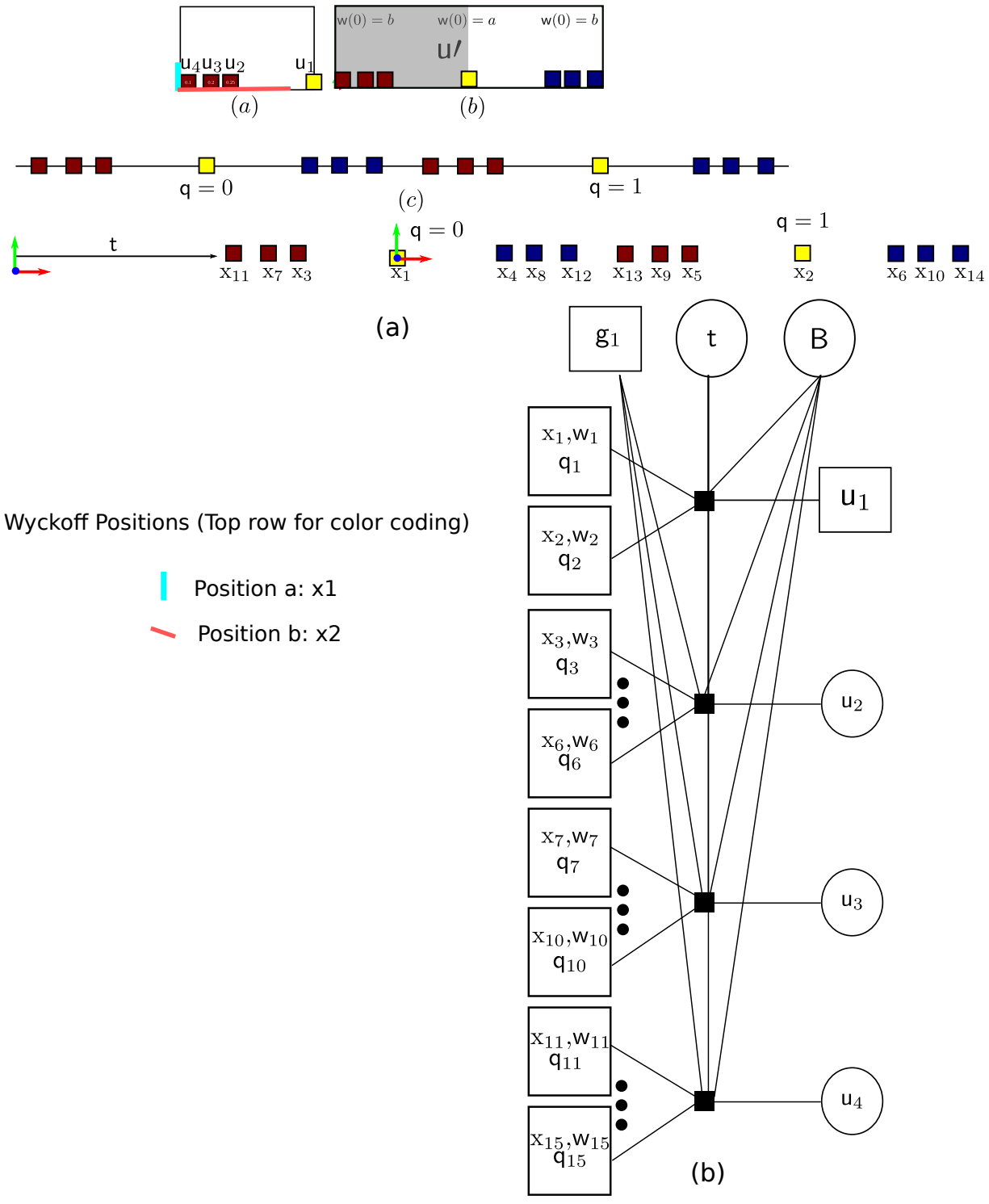


Figure 3.4: **Factor graph for the optimizing points in a 1D lattice and 1D symmetry**. We are given a set of 12 measurements of point $\mathcal{X} \in \mathsf{x}_1 \ldots \mathsf{x}_{12}$ and its Wyckoff indicator variables $\mathcal{W} = \{w_1 \ldots w_{12}\}$ and its Miller indices $\mathcal{Q} = \{\mathsf{q}_1 \ldots \mathsf{q}_{12}\}$. Our goal is to estimate the unknown location of points in the asymmetric unit $\mathcal{U} = \{\mathsf{u}_1, \mathsf{u}_2, \mathsf{u}_3\}$ that generate these 12 points $\mathcal{X}$ along with the basis of the lattice B and the translation vector t. The generator g_1 corresponds to reflection about the vertical axis passing through the origin, same as the previous case!

1D Lattice

In this section, I develop the generative model for generating 1D points that are symmetric and exhibit a lattice structure. Figure 3.3(a) shows an example of this configuration.

In addition to this, since this symmetry now exhibits a lattice structure, there are two additional parent variables that need to modeled (1) the miller indices represented by $\mathsf{q} \in \mathbb{Z}$ and the basis of the lattice represented by $\mathsf{B} \in \mathbb{R}^{+}$.

The joint probability distribution that this generative model describes is simply given by the single conditional distribution

$$p\left(\Theta\right) \triangleq p\left(\mathcal{X} \middle| \mathcal{U}, \mathsf{g}_1, \mathcal{W}, \mathsf{t}, \mathsf{B}, \mathcal{Q}\right) \tag{3.11}$$

where $\mathcal{X} = \{\mathsf{x}_j\}_{j=1}^{n} \Big| \mathsf{x}_j \in \mathbb{R}$ is the set of points that are symmetric. $\mathcal{U} = \{\mathsf{u}_p\}_{p=1}^{P} \Big| \mathsf{u}_p \in -\infty, 0]$ are the set of points in the asymmetric unit, $\mathcal{W} = \{\mathsf{w}_p\}_{j=1}^{n} \Big| \mathsf{w} \in \{\{a, b\}, \{0, 1\}\}$ are the set of Wyckoff position for every point in $\mathcal{X}$ and $\mathsf{g}_1 = -1$ is the generator corresponding to reflection,$\mathsf{t} \in \mathbb{R}$ is the translation vector, $\mathsf{B} \in \mathbb{R}^{+}$ is the basis matrix, a scalar in this case that takes only positive values and the Miller indices $\mathsf{q} \in \mathcal{Q} \Big| \mathsf{q} \in \mathbb{Z}$ for every point in $\mathcal{X}$. We follow the same modeling procedure for the index of the variables inside the asymmetric unit and its relation to the observed points, i.e $\mathsf{u}_{\mathbf{p}_j} \rightarrow \mathsf{x}_j$.

We can represent the generative model by means of a measurement function given by

$$\mathsf{x}_j = h_{11}\left(\mathsf{u}_p, \mathsf{g}_1, \mathsf{q}_j, \mathsf{w}_j, \mathsf{B}, \mathsf{t}\right) + \eta_{11} \tag{3.12}$$

where $\eta_{10} = \mathcal{N}(\mathsf{x}, 0, \sigma)$ is a Gaussian-corrupted noise on the point x on the symmetric structure with a covariance σ(same as above). The measurement function now models the added lattice parameters q_j and the basis B. The measurement function $h_{10}(\mathsf{v}_p, \mathsf{g}_1, \mathsf{w}_j, \mathsf{t})$ is given by

$$h_{11}(\mathsf{u}_p, \mathsf{g}_1, \mathsf{q}_j, \mathsf{B}, \mathsf{w}_j, \mathsf{t}) \equiv \begin{cases} \mathsf{B}\left[\mathsf{g}_1^{\mathsf{w}(1)}\mathsf{u}_p + \mathsf{q}_j\right] + \mathsf{t} & \mathsf{w}(0) = b \\ \mathsf{B}\mathsf{q}_j + \mathsf{t} & \mathsf{w}(0) = a \end{cases} \tag{3.13}$$

which describes process to generate a point $\mathsf{x}_j \in \mathbb{R}$ in a 1D lattice setting shown in Figure 3.3. We start from its corresponding point inside the asymmetric unit u_{p_j}. If the point is in a general position $(\mathsf{w}(0) = b)$ (shown in maroon in the figure) it has the highest multiplicity ($\times 2$) and can belong to either the unreflected region or the reflected region of the unit cell. However if the point is in the special position (the point shown in yellow) its multiplicity is 1 and this corresponds to the location of the line of reflection inside the unit asymmetric unit. Since for this example, this is chosen as the origin, $\mathsf{u}_{p_j} = 0$.

If we know the location of point in the general position only, Equation 3.13 reduces to

$$h_{11}(\mathsf{u}_p, \mathsf{g}_1, \mathsf{q}_j, \mathsf{B}, \mathsf{w}_j, \mathsf{t}) = \mathsf{B}\left[\mathsf{g}_1^{\mathsf{w}(1)}\mathsf{u}_p + \mathsf{q}_j\right] + \mathsf{t} \tag{3.14}$$

The likelihood function, a function of the parameters of the model $\mathcal{U}, \mathsf{t}, \mathcal{W}, \mathsf{B}, \mathcal{Q}, \mathsf{g}_1, \mathsf{P}$ and the observed data x_j is given by

$$\mathcal{L}(\mathcal{X}; \mathcal{U}, \mathsf{t}, \mathcal{W}, \mathsf{g}_1, \mathsf{B}, \mathcal{Q}) \propto \prod_{j=1}^{n} \exp\left\| \frac{\left(\mathsf{x}_j - \mathsf{x}_j^{\dagger}\right)^2}{2\sigma} \right\| \tag{3.15}$$

where the $\mathsf{x}_j^\dagger$ is the predicted location of the point using the measurement prediction function given in Equation 3.13 and x_j is the observed location (notice the extra variables as compared to the previous case). We can once again obtain the optimal value by determining the value of the unknown variables $\mathcal{U}, \mathsf{t}, \mathsf{B}$ that maximizes the negative log likelihood.

$$(\mathcal{U}^\star, \mathsf{B}^\star, \mathsf{t}^\star) = \arg\max_{\mathcal{U},\mathsf{B},\mathsf{t}} \left(\frac{1}{2\sigma^2} \sum_{j=1}^{n} \left\| \mathsf{x}_j - \mathsf{x}_j^\dagger \right\|^2 \right) \tag{3.16}$$

Figure 3.4 shows a factor graph that corresponds to this optimization problem for a small example setup. We show points in both Wyckoff positions (a, b) and show how they multiply through the lattice and the asymmetric units generating the all the points in $\mathcal{X}$. The example optimization shown here consists of 12 points generated from 4 points in the asymmetric unit. If we know the Wyckoff positions and generators, we can immediately assign a known location to point $\mathsf{u}_4 = 0$.

Generating Symmetries in 2D

In this section we will look into the generation of symmetric points in 2D. Since we have laid the foundation of the notation framework in Section 3.1, I restrict this section to the additional parameters that arise as a consequence of 2D space. Similar to the case of 1D symmetries, I proceed by first discussing the case of no lattice in 2D symmetry followed by the two cases of 1D and 2D lattices.

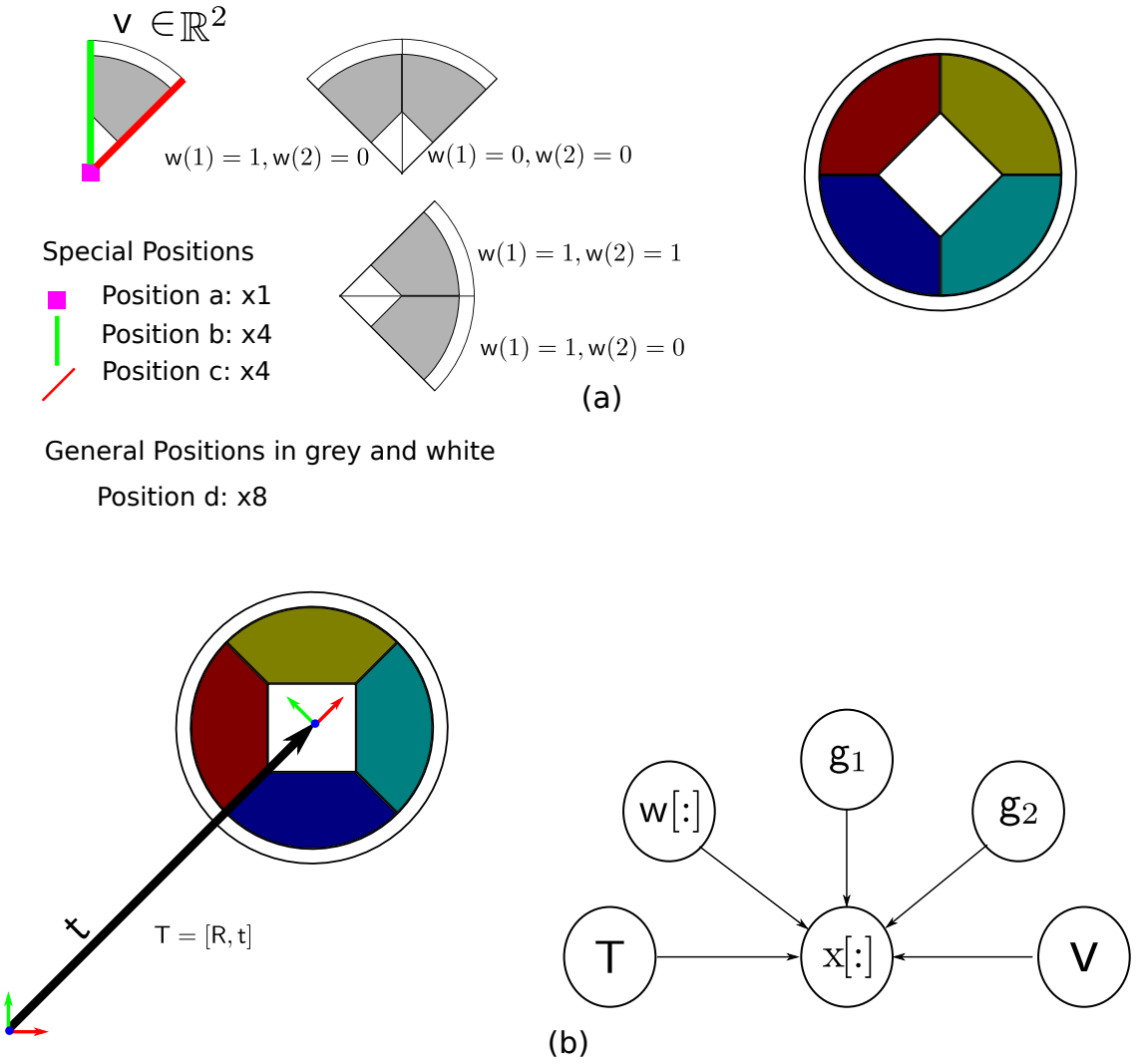


Figure 3.5: **Generative model for synthesizing 2D symmetry with no lattice**. (a) I show an axial symmetry group in 2D, the dihedral group D_4. This group is a combination of two types of symmetries the bilateral symmetry and a rotational symmetry. The symmetry in its canonical coordinate frame is shown in the top right. The entire symmetric structure is translated and rotated as shown in bottom left. The asymmetric unit is shown in the top left, it a specific segment of the circle. There are 4 Wyckoff positions, the lowest order is at the center of the circle and has a multiplicity of 1. There are two positions along the edges of the segment along the radii that each have a multiplicity of 4. The general position has multiplicity of 8. $\mathsf{v} \in \mathbb{R}^2$, $\mathsf{x} \in \mathbb{R}^2$ and the Wyckoff indicator variables spans $\mathsf{w} = \{\{a, b, c, d\}, \{0, 1\}, \{0, 1, 2, 3\}\}$. (b) The generative model for this case is similar to the 1D case with no lattice, with an additional generator g_2 corresponding to rotation and the simple translation t replaced by a full pose $\mathsf{T} \in \mathbb{SE}(2)$consisting of both rotation and translation.

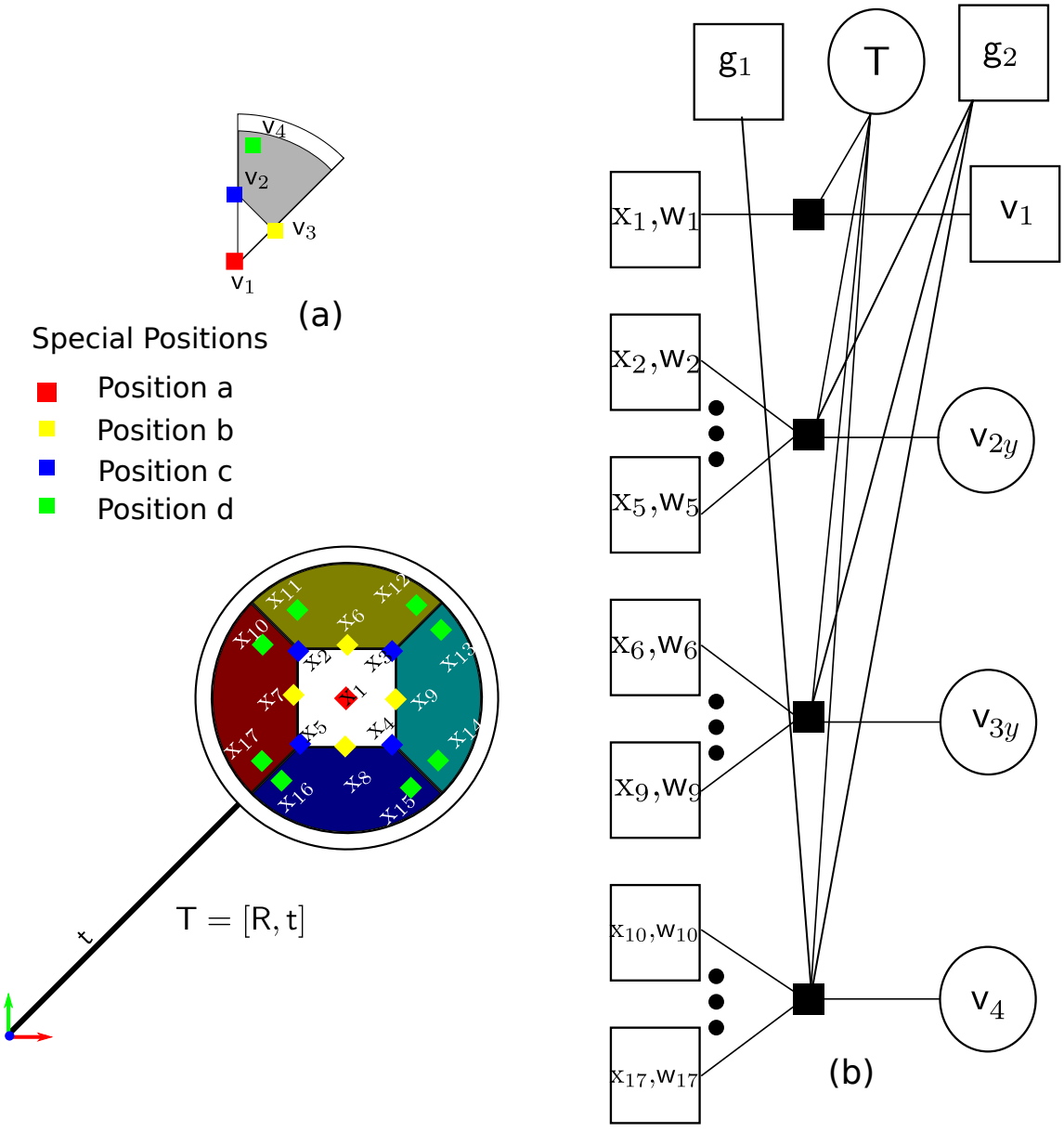


Figure 3.6: **Factor graph for the optimization problem of estimating 2D symmetric points in a no-lattice setting**: We are given a set of 17 measurements of point $\mathcal{X} \in \mathsf{x}_1 \dots \mathsf{x}_{17}$ and its Wyckoff indicator variables $\mathcal{W} = \{w_1 \dots w_{17}\}$. Our goal is to estimate the location of points in the asymmetric unit $\mathcal{V} = \{\mathsf{v}_1 \dots \mathsf{v}_4\}$ that generate these 17 points $\mathcal{X}$. The generator g_1 corresponds to reflection about the vertical axis passing through the origin and the generator g_2 corresponds to a rotations of 90^0 about the origin, $\mathsf{T} \in \mathbb{SE}(2)$ is the rotation and translation of the symmetric structure relative to a world origin. v_1 is immediately known as $(0, 0)$ is we choose the location of the canonical coordinate frame at the center of the circle. $\mathsf{v}_2 = \begin{bmatrix} 0 \\ \mathsf{v}_{2y} \end{bmatrix}$ and $\mathsf{v}_3 = \mathsf{v}_{3k} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ is a point on the line $x + y = 1$.

No Lattice

In this section, I develop the generative model for generating 2D points that are symmetric and do not exhibit any lattice structure. Figure 3.5 shows an example of this configuration.

The generative model that corresponds to this specific instance of symmetry is shown in Figure 3.3(b). The points in 2D x span the space $\mathbb{R}^2$. Every point has a Wyckoff position that spans a $3-$tuple $\mathsf{w} \in \{\{a,b,c,d\},\{0,1\},\{0,1,2,3\}\}$ where the first element of the tuple specifies the Wyckoff position $\{a,b,c,d\}$ which is depicted in Figure 3.3(a), the second element specifies if the point is reflected or unreflected with respect the the y-axis and the thirst element spans the set of integers $\{1,2,3,4\}$ specifying the order of rotation. The points inside the asymmetric unit are given by $\mathcal{V}$. The generator that models the reflection along the $y-$axis is given by

$$\mathsf{g}_1 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \tag{3.17}$$

and the second generator that models a rotation of 90^0 counter clockwise is given by

$$\mathsf{g}_2 = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \tag{3.18}$$

Every point x has a corresponding location inside the asymmetric unit. This location is described by the variable $\mathsf{v} = \begin{bmatrix} \mathsf{v}_x \\ \mathsf{v}_y \end{bmatrix} \Big| \mathsf{v}_x + \mathsf{v}_y \leq 1, \sqrt{\mathsf{v}_x^2 + \mathsf{v}_y^2} \leq R, \mathsf{v}_x \geq 0$ for our choice of origin and reflection planes. In addition to this, we allow a rigid transformation of the

entire symmetric structure from an arbitrarily assigned world coordinate frame we call this translation $\mathsf{T} \in \mathbb{SE}(2)$.

The joint probability distribution has the additional generator g_2 as compared to the 1D, no lattice case

$$p(\Theta) \triangleq p\left(\mathcal{X} \middle| \mathcal{V}, \mathsf{g}_1, \mathsf{g}_2, \mathcal{W}, \mathsf{T}\right) \tag{3.19}$$

where $\mathcal{X} = \{\mathsf{x}_j\}_{j=1}^{n} \Big| \mathsf{x}_j \in \mathbb{R}^2$ is the set of points that are symmetric. $\mathcal{V} = \{\mathsf{v}_p\}_{p=1}^{P} \Big| \mathsf{v}_p = \begin{bmatrix} \mathsf{v}_{px} \\ \mathsf{v}_{py} \end{bmatrix} \Big| \mathsf{v}_{px} + \mathsf{v}_{py} \leq 1, \sqrt{\mathsf{v}_{px}^2 + \mathsf{v}_{py}^2} \leq R, \mathsf{v}_{px} \geq 0$ are the set of points in the asymmetric unit, $\mathcal{W} = \{\mathsf{w}_p\}_{j=1}^{n} \Big| \mathsf{w} \in \{\{a, b, c, d\}, \{0, 1\}, \{0, 1, 2, 3\}\}$ are the set of Wyckoff position for every point in $\mathcal{X}$ and $\mathsf{g}_1, \mathsf{g}_2$ are given above and $\mathsf{T} = (\mathsf{R}, \mathsf{t}) \in \mathbb{SE}(2)$. The correspondence modeling still holds from the case of 1D, i.e $\mathsf{v}_{\mathbf{p}_j} \rightarrow \mathsf{x}_j$

We can represent the generative model by means of a measurement function given by

$$\mathsf{x}_j = h_{20}(\mathsf{v}_p, \mathsf{g}_1, \mathsf{g}_2, \mathsf{w}_j, \mathsf{T}) + \eta_{20} \tag{3.20}$$

where $\eta_{20} = \mathcal{N}(\mathsf{x}, 0, \Sigma)$ is a Gaussian-corrupted noise on the point x on the symmetric structure with a covariance $\Sigma \in \mathbb{R}^{2\times 2}$. The measurement function $h_{20}(\mathsf{v}_p, \mathsf{g}_1, \mathsf{g}_2, \mathsf{w}_j, \mathsf{T})$ is

given by

$$h_{20}\left(\mathsf{v}_p, \mathsf{g}_1, \mathsf{g}_2, \mathsf{w}_j, \mathsf{T}\right) \equiv \begin{cases} \mathsf{R}\left(\mathsf{g}_1^{\mathsf{w}_j(1)} \mathsf{g}_2^{\mathsf{w}_j(2)} \mathsf{v}_{p_j}\right) + \mathsf{t} & \mathsf{w}\left(0\right) = d \\ \mathsf{R}\left(\mathsf{g}_2^{\mathsf{w}_j(2)} \begin{bmatrix} 0 \\ \\ \mathsf{v}_{p_j y} \end{bmatrix}\right) + \mathsf{t},\ 0 \leq \mathsf{v}_{p_j y} \leq R & \mathsf{w}\left(0\right) = c \\ \mathsf{R}\left(\mathsf{g}_2^{\mathsf{w}_j(2)} \mathsf{v}_{p_j y} \begin{bmatrix} 1 \\ \\ 1 \end{bmatrix}\right) + \mathsf{t},\ 0 \leq \mathsf{v}_{p_j y} \leq R & \mathsf{w}\left(0\right) = b \\ \mathsf{t} & \mathsf{w}\left(0\right) = a \end{cases} \tag{3.21}$$

which describes the following process, to generate a point $\mathsf{x}_j \in \mathbb{R}^2$ shown in Figure 3.6: we start from its corresponding point inside the asymmetric unit v_{p_j}. If the point is in a general position $\left(w\left(0\right) = d\right)$ (shown in green) it has the highest multiplicity ($\times 8$) and is affected by both the generators $\mathsf{g}_1, \mathsf{g}_2$. If a point is at the center of the circle (shown in red), its location is immediately know as $\mathsf{v}_{p_j} = (0, 0)$. If a point is located along one of the two edges of the segment (shown in blue and yellow), it is only affected by the generator corresponding to rotation g_2. Its multiplicity is 4.

If we know the location of point in the general position only, Equation 3.21 reduces to

$$h_{20}\left(\mathsf{v}_p, \mathsf{g}_1, \mathsf{g}_2, \mathsf{w}_j, \mathsf{T}\right) = \mathsf{R}\left(\mathsf{g}_1^{\mathsf{w}_j(1)} \mathsf{g}_2^{\mathsf{w}_j(2)} \mathsf{v}_{p_j}\right) + \mathsf{t} \tag{3.22}$$

Figure 3.6(b) shows a factor graph that corresponds to this optimization problem for a small example setup. We have chosen the location of the points at different Wyckoff positions to show how the corresponding unknown factor would look like. The point in the

general position is given by $\mathsf{v}_4 \in \mathbb{R}^2 \Big| \mathsf{v}_4 = \begin{bmatrix} \mathsf{v}_{4x} \\ \mathsf{v}_{4y} \end{bmatrix} \Big| \mathsf{v}_{4x} + \mathsf{v}_{4y} < 1, \sqrt{\mathsf{v}_{4x}^2 + \mathsf{v}_{4y}^2} \leq R, \mathsf{v}_{4x} > 0$ and the point in the special positions v_2 and v_3 are an optimization of scalar value that measured the distance from the center of the circle to the point denoted by $0 < \mathsf{v}_{2y} \leq R$ and $0 < \mathsf{v}_{3y} \leq R$. The point at the center v_1 is known as $(0, 0)$.

1D Lattice

This case is the simplest case of a sub-dimensional lattice. I will use a slightly modified version of the example given in Section 3.1.1 to mainly elucidate how sub-dimensional lattices form a part of my generative model. In this section, I develop the generative model for generating 2D points that are symmetric and exhibit a 1D lattice structure. Figure 3.3(a) shows an example of this configuration.

Similar to the 1D lattice case, we model the lattice related variables as (1) the Miller indices represented by $\mathsf{q} \in \mathbb{Z}$ and the basis of the lattice represented by $\mathsf{B} \in \mathbb{R}^+$. The joint probability distribution that this generative model describes is simply given by the single conditional distribution

$$p(\Theta) \triangleq p\left(\mathcal{X} \middle| \mathcal{U}, \mathcal{V}, \mathsf{g}_1, \mathcal{W}, \mathsf{T}, \mathsf{B}, \mathcal{Q}\right) \tag{3.23}$$

where $\mathcal{X} = \{\mathsf{x}_j\}_{j=1}^n \Big| \mathsf{x}_j \in \mathbb{R}^2$ is the set of points that are symmetric. $\mathcal{U} = \{\mathsf{u}_p\}_{p=1}^P \Big| \mathsf{u}_p \in -\infty, 0]$ and $\mathcal{V} = \{\mathsf{v}_p\}_{p=1}^P \Big| \mathsf{v}_p \in \mathbb{R}$ is projections of the points along a direction that is perpendicular to the direction of the 1D lattice. Since we have chosen the same example

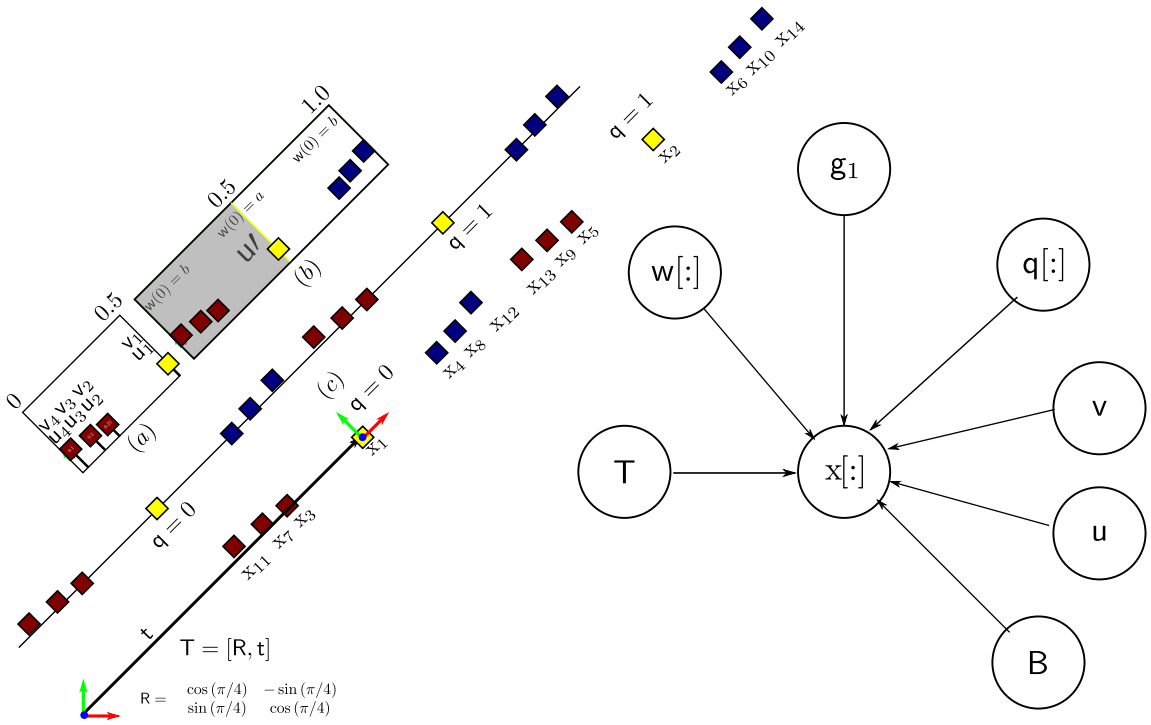


Figure 3.7: **Example of Symmetry synthesis for 1D lattice in 2D space**: using the generative model shown in Figure 3.7. (a) show three points inside the asymmetric unit in a canonical coordinate frame for reflection symmetry. (b) The generated unit cell. All the points are in a general position and therefore have a multiplicity of $\times 2$. (c) represents the points in the lattice and (d) represents the entire lattice translated by an unknown t to generate the final set of points $\mathcal{X}$

as in Section 3.1.1, the Wyckoff positions and assosiated generator action modeled by $\mathcal{W}$ spans the same space, $\mathcal{W} = \{\mathsf{w}_j\}_{j=1}^{n} \Big| \mathsf{w} \in \{\{a, b\}, \{0, 1\}\}$. However, the generator

$$\mathsf{g}_1 = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \tag{3.24}$$

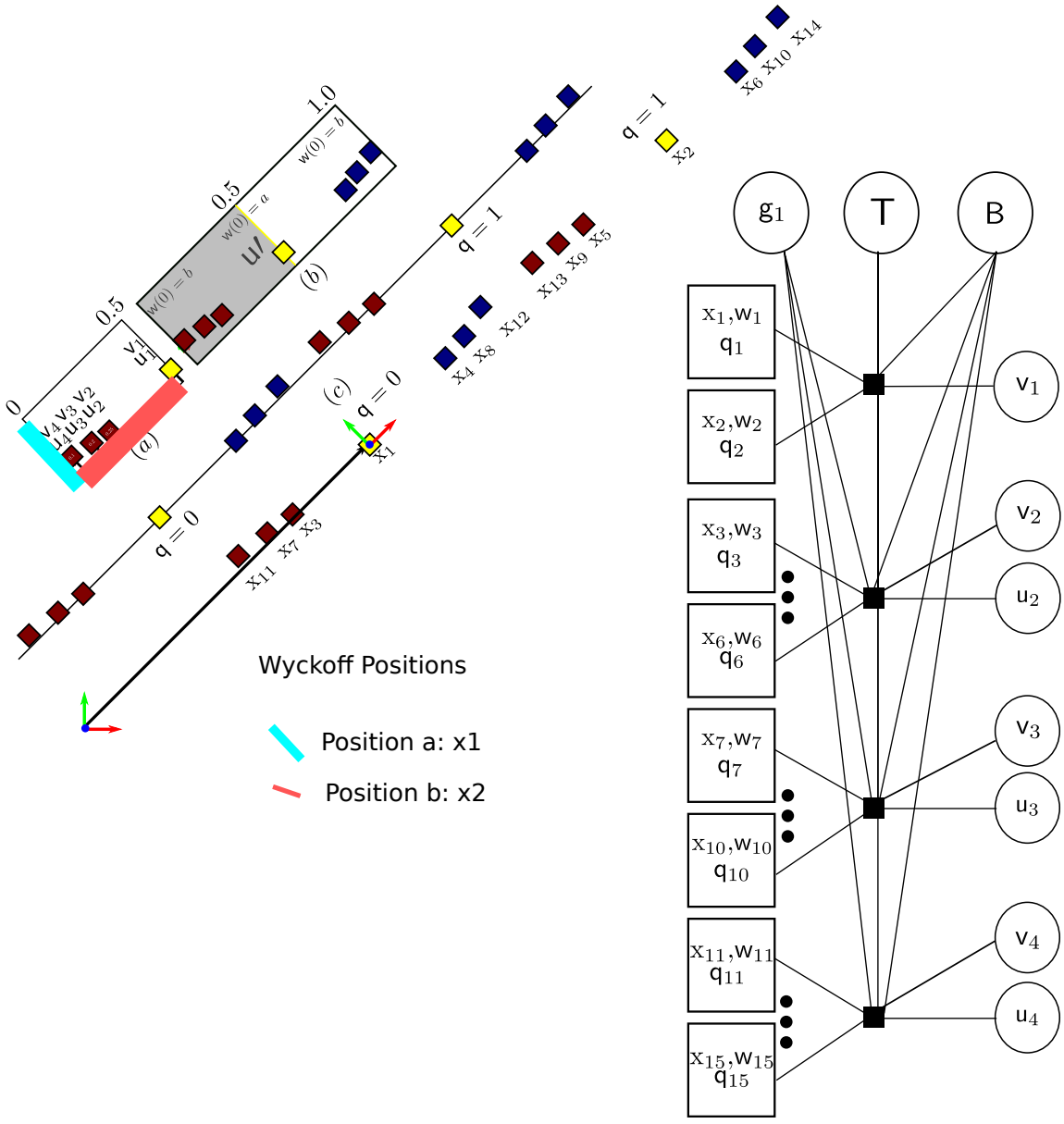


Figure 3.8: **Factor graph and example optimization for 2D lattice with 2D symmetry**: using the generative model shown in Figure 3.9. (a) show three points inside the asymmetric unit in a canonical coordinate frame for reflection symmetry. (b) The generated unit cell. All the points are in a general position and therefore have a multiplicity of $\times 2$. (c) represents the points in the lattice and (d) represents the entire lattice translated by an unknown t to generate the final set of points $\mathcal{X}$

acts upon a 2D point $\begin{bmatrix} \mathsf{u}_p \\ \mathsf{v}_p \end{bmatrix}$ models the reflection parallel to the axis on which we chose to represent $\mathcal{V}$ with a similar choice of coordinate system as in the case of 1D. $\mathsf{T} \in \mathbb{SE}(2)$, $\mathsf{B} \in \mathbb{R}^{+}$,$\mathsf{q} \in \mathcal{Q} \Big| \mathsf{q} \in \mathbb{Z}$ and $\mathsf{v}_{\mathbf{p}_j}, \mathsf{u}_{\mathbf{p}_j} \rightarrow \mathsf{x}_j$.

We can represent the generative model by means of a measurement function given by

$$\mathsf{x}_j = h_{21}\left(\mathsf{v}_p, \mathsf{u}_p, \mathsf{g}_1, \mathsf{q}_j, \mathsf{w}_j, \mathsf{B}, \mathsf{T}\right) + \eta_{21} \tag{3.25}$$

where $\eta_{21} = \mathcal{N}\left(\mathsf{x}, 0, \Sigma\right)$ is a Gaussian-corrupted noise on the point x on the symmetric structure with a covariance $\Sigma \in \mathbb{R}^{2\times 2}$.(same as above). The measurement function now models the added lattice parameters q_j and the basis B. The measurement function $h_{21}\left(\mathsf{v}_p, \mathsf{u}_p, \mathsf{g}_1, \mathsf{q}_j, \mathsf{w}_j, \mathsf{B}, \mathsf{T}\right)$ is given by

$$h_{21}\left(\mathsf{v}_p, \mathsf{u}_p, \mathsf{g}_1, \mathsf{q}_j, \mathsf{w}_j, \mathsf{B}, \mathsf{T}\right) \equiv \begin{cases} \mathsf{B}\left[\mathsf{g}_1^{\mathsf{w}(1)}\begin{bmatrix} \mathsf{u}_p \\ \mathsf{v}_p \end{bmatrix} + \mathsf{q}_j \begin{bmatrix} 1 \\ 0 \end{bmatrix}\right] + \mathsf{t} & \mathsf{w}\left(0\right) = b \\ \mathsf{B}\left[\mathsf{g}_1^{\mathsf{w}(1)}\begin{bmatrix} 0 \\ \mathsf{v}_p \end{bmatrix} + \mathsf{q}_j \begin{bmatrix} 1 \\ 0 \end{bmatrix}\right] + \mathsf{t} & \mathsf{w}\left(0\right) = a \end{cases} \tag{3.26}$$

which describes process to generate a point $\mathsf{x}_j \in \mathbb{R}$ in a 1D lattice setting shown in Figure 3.3. We start from its corresponding point inside the asymmetric unit $\mathsf{u}_{p_j}, \mathsf{v}_{p_j}$. If the point is in a general position $\left(\mathsf{w}\left(0\right) = b\right)$ (shown in maroon in the figure) it has the highest multiplicity ($\times 2$) and can belong to either the unreflected region or the reflected region of the unit cell. However if the point is in the special position (the point shown in yellow)

its multiplicity is 1 and this corresponds to the location of the line of reflection inside the asymmetric unit. Since for this example, this is chosen as the origin, $\mathsf{u}_{p_j} = 0$. This argument is almost exactly same as the 1D lattice case. The only addition being the unknown projection v_{p_j}.

If we know the location of point in the general position only, Equation 3.26 reduces to

$$h_{21}\left(\mathsf{v}_p, \mathsf{u}_p, \mathsf{g}_1, \mathsf{q}_j, \mathsf{w}_j, \mathsf{B}, \mathsf{T}\right) = \mathsf{B}\left[\mathsf{g}_1^{\mathsf{w}(1)}\begin{bmatrix}\mathsf{u}_p \\ \mathsf{v}_p\end{bmatrix} + \mathsf{q}_j\begin{bmatrix}1 \\ 0\end{bmatrix}\right] + \mathsf{t} \tag{3.27}$$

Figure 3.4 shows a factor graph that corresponds to this optimization problem for a small example setup. We show points in both Wyckoff positions (a, b) and show how they multiply through the lattice and the asymmetric units generating the all the points in $\mathcal{X}$. The example optimization shown here consists of 14 points generated from 4 points in the asymmetric unit.

2D Lattice

We now consider the generative process of a complex 2D symmetry with 2D lattice. Figure 3.9 shows a generative model of the process of symmetry synthesis in 2D. The group that we are going to choose for our example is the wallpaper group $c2mm$. From the crystallographic table, we can see that this group has 5 centers of two-fold rotation located at $\{(0.25, 0.25), (0.75, 0.25), (0.25, 0.75), (0.75, 0.75), (0.5, 0.5)\}$ and mutually perpendicular lines of reflection at 0.5 and 0.5 along the x and y axis. The general position of a point in this group has a multiplicity of 8. There are 5 special positions inside the unit cell:

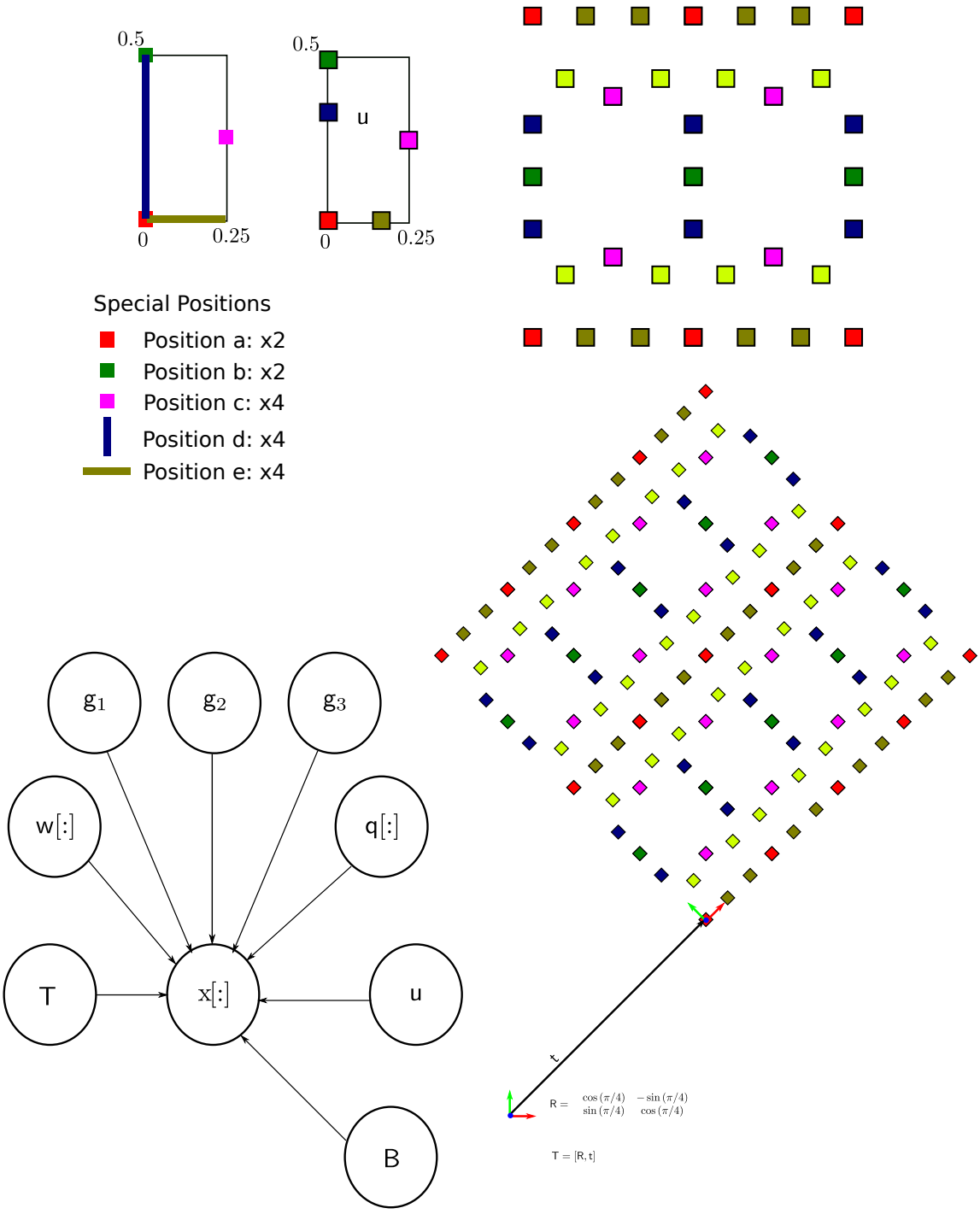


Figure 3.9: **Example of symmetry synthesis for 2D lattice in a 2D symmetry**. The generative model for the 2D wallpaper group $c2mm$. [Top left] we show six points inside the asymmetric unit in a canonical coordinate frame and I also show the general and special position inside the asymmetric unit. Among the 6 points, there is only one point in the general position. [Top right] I show the unit cell of the lattice. [Bottom Right] I show 4 unit cells corresponding to Miller indices $(0,0),(0,1)(1,0),(1,1)$. [Bottom Right] The generative model for this particular symmetry group has three generators g_1, g_2, g_3 which we explain in detail in Section 3.2.2.

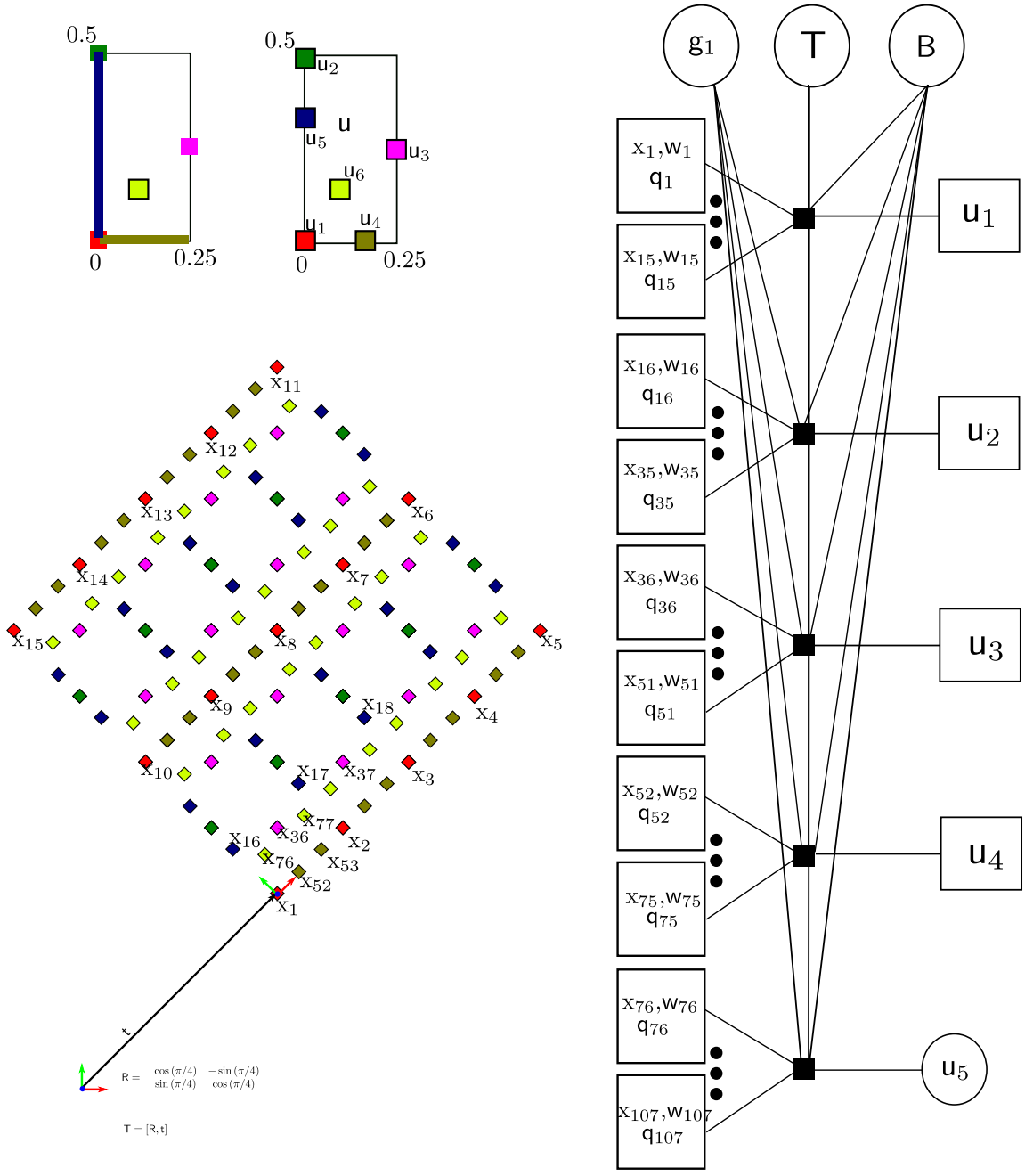


Figure 3.10: **Factor graph and example optimization for 2D lattice with 2D symmetry**: Here we show an example problem for optimizing for the location of 6 points in the asymmetric unit $(\mathsf{u}_1 \dots \mathsf{u}_6)$. Figure 3.9 shows the corresponding generative model. For generating 4 unit cells of the lattice, the 6 points are multiplies into 107 points. We follow a counterclockwise, color coded sequential numbering for the points, some of these numberings are shown in the bottom left diagram. On the right we have the factor graph, its Wyckoff indicator variables $\mathcal{W} = \{w_1 \dots w_{107}\}$. Our goal is to estimate the location of points in the asymmetric unit $\mathcal{U} = \{\mathsf{u}_1 \dots \mathsf{u}_6\}$ that generate these 107 points $\mathcal{X}$. The generators $\mathsf{g}_1, \mathsf{g}_2, \mathsf{g}_3$ are explained in detail below and the transformation of the lattice relative to a world coordinate frame is $\mathsf{T} \in \mathbb{SE}(2)$.

1. **Position a** : This corresponds to the following location in the asymmetric unit $\mathsf{u} = (0, 0)$. The multiplicity of point here is two because the point at $(0, 0)$ is reflected by planes at $\mathsf{u}_x = 0.5$ and $\mathsf{u}_y = 0.5$ to generate points as $(1, 0)\,, (1, 1)\,, (0, 1)$. However, we can count only $1/4^{th}$ of all these points because they border 4 adjacent unit cells (including $(0, 0)$), so this amount to a total point of 1 inside the unit cell. Additionally the point $(0, 0)$ can be rotated by the center of rotation at $(1/4, 1/4)$ and then reflected by either by $\mathsf{u}_x = 0.5$ or $\mathsf{u}_y = 0.5$ to create the point $(1/2, 1/2)$. Thus the multiplicity of the point is 2. However while the table shows two special position inside the unit cell, there is only one special position corresponding position inside the asymmetric unit and this is $(0, 0)$. The site symmetry of the group is $2mm$ which corresponds to three generators

$$\mathsf{g}_1 = \begin{bmatrix} -1 & 0 & 0.25 \\ 0 & -1 & 0.25 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.28}$$

for the two fold rotation at $(1/4, 1/4)$ and

$$g_2 = \begin{bmatrix} -1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.29}$$

for reflection with respect to the line $u_x = 0.5$ and

$$g_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.30}$$

for reflection with respect to the line $u_y = 0.5$.

2. **Position b** : This corresponds to the following location in the asymmetric unit $\mathsf{u} = (0, 1/2)$. This is the point at the left center of the unit cell. This can be reflected by the line $u_y = 0.5$ to generate the point $(1, 1/2)$. Both these points border one more adjacent unit cell the count to a total of 1. In addition, as in the previous case, the point can be rotated about $(1/4, 1/4)$ to generate the point $(1/2, 0)$ which can be reflected around $u_x = 0.5$ to generate the points $(1/2, 0)$ and $(1/2, 1)$. So the total multiplicity of the point is 2.

3. **Position c** : This corresponds to the location $(\mathsf{u} = (1/4, 1/4))$ inside the symmetric unit and this has a multiplicity of 4. We can see this right away because reflecting at about the lines $u_x = 0.5$ and $u_y = 0.5$ generates the points $(1/4, 3/4)$, $(3/4, 1/4)$, $(3/4, 3/4)$. Which leads to a total of 4 points including the one at $(1/4, 1/4)$.

4. **Position d :** This corresponds to any location in the asymmetric unit such that $\mathsf{u}\Big|\mathsf{u}_y = 0$ and $\mathsf{u}_x \neq (1/4), \mathsf{u}_x \neq (0)$ which is the bottom border of the asymmetric unit. Any point here has a multiplicity of 4. Once again we can analyze this by using the three generators above. This leads to following regions in the image $\mathsf{u} = (\mathsf{u}_x, 0)\Big|\mathsf{u}_x \neq (1/2), \mathsf{u}_x \neq (0)$, $\mathsf{u} = (1 - \mathsf{u}_x, 0)\Big|\mathsf{u}_x \neq (1/2), \mathsf{u}_x \neq (0)$,

$\mathsf{u} = (1 - \mathsf{u}_x, 1) \left| \mathsf{u}_x \neq (1/2), \mathsf{u}_x \neq (0), \mathsf{u} = (\mathsf{u}_x + 1/4, 1/2) \right| \mathsf{u}_x \neq (1/2), \mathsf{u}_x \neq (0),$
$\mathsf{u} = (1 - (\mathsf{u}_x + 1/4), 1/2) \left| \mathsf{u}_x \neq (1/2), \mathsf{u}_x \neq (0). \right.$

5. **Position e** : This corresponds to s to any location in the asymmetric unit such that $\mathsf{u} \left| \mathsf{u}_x = 0 \right.$ and $\mathsf{u}_y \neq (1/2), \mathsf{u}_y \neq (0)$. This is the left border of the asymmetric unit and can be analyzed using the above generators.

Therefore, to generate any point x in this group $c2mm$, we need the corresponding point inside the asymmetric unit $\mathsf{u} \left| \mathsf{0} \leq \mathsf{u}_x \leq \mathsf{1/4}, \mathsf{0} \leq \mathsf{u}_y \leq \mathsf{1/2} \right.$ and the binary vector $\mathsf{w} \in [\{a, b, c, d, e, f\} \{0, 1, 2, 3\}, \{0, 1\}, \{0, 1\}]$. We only need need 3 generators to generate all the points in this system.

A generative model for the group $c2mm$ is given by

$$\mathsf{x} = \begin{cases} \mathsf{RB}\left[\mathsf{g}_1^{\mathsf{w}(1)} \circ \mathsf{g}_2^{\mathsf{w}(2)} \circ \mathsf{g}_3^{\mathsf{w}(3)} \mathsf{u} + \mathsf{q}\right] + \mathsf{t} & \mathsf{w}\left(0\right) = f \\ \mathsf{RB}\left[\mathsf{g}_1^{\mathsf{w}(1)} \circ \mathsf{g}_2^{\mathsf{w}(2)} \circ \mathsf{g}_3^{\mathsf{w}(3)} \begin{bmatrix} 0 \\ \mathsf{u}_y \end{bmatrix} + \mathsf{q}\right] + \mathsf{t} & \mathsf{w}\left(0\right) = e \\ \mathsf{RB}\left[\mathsf{g}_1^{\mathsf{w}(1)} \circ \mathsf{g}_2^{\mathsf{w}(2)} \circ \mathsf{g}_3^{\mathsf{w}(3)} \begin{bmatrix} \mathsf{u}_x \\ 0 \end{bmatrix} + \mathsf{q}\right] + \mathsf{t} & \mathsf{w}\left(0\right) = d \\ \mathsf{RB}\left[\mathsf{g}_1^{\mathsf{w}(1)} \circ \mathsf{g}_2^{\mathsf{w}(2)} \circ \mathsf{g}_3^{\mathsf{w}(3)} \begin{bmatrix} 0.25 \\ 0.25 \end{bmatrix} + \mathsf{q}\right] + \mathsf{t} & \mathsf{w}\left(0\right) = c \\ \mathsf{RB}\left[\mathsf{g}_1^{\mathsf{w}(1)} \circ \mathsf{g}_2^{\mathsf{w}(2)} \circ \mathsf{g}_3^{\mathsf{w}(3)} \begin{bmatrix} 0 \\ 0.5 \end{bmatrix} + \mathsf{q}\right] + \mathsf{t} & \mathsf{w}\left(0\right) = b \\ \mathsf{RB}\left[\mathsf{g}_1^{\mathsf{w}(1)} \circ \mathsf{g}_2^{\mathsf{w}(2)} \circ \mathsf{g}_3^{\mathsf{w}(3)} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \mathsf{q}\right] + \mathsf{t} & \mathsf{w}\left(0\right) = a \end{cases} \tag{3.31}$$

where $\mathsf{t} \in \mathbb{R}^2$, $\mathsf{B} \in \mathbb{R}^{2\times 2}$, $\mathsf{R} \in \mathbb{SO}\left(2\right)$, the system of generators $g = \{g_1, g_2, g_3\}$ are defined above, $0 \leq \mathsf{u}_x \leq \frac{1}{4}$, $0 \leq \mathsf{u}_y \leq \frac{1}{2}$ and $\mathsf{q} \in \mathbb{Z}^2$. This equation describes the process to generate a point $\mathsf{x}_j \in \mathbb{R}$ in a 1D lattice setting shown in Figure 3.10. We start from its corresponding point inside the asymmetric unit u_{p_j}. If the point is in a general position $\left(\mathsf{w}\left(0\right) = f\right)$ (shown in green in the figure) it has the highest multiplicity ($\times 8$) and each of these points is shown in Figure 3.3. However if the point is in the special position, we have specific constraints on it just like in the previous cases. These constraints are graphically shown in Figure 3.10 and in Equation 3.31

If we know the location of point in the general position only, Equation 3.31 reduces to

$$\mathsf{x} = \mathsf{RB}\left[\mathsf{g}_1^{\mathsf{w}(1)} \circ \mathsf{g}_2^{\mathsf{w}(2)} \circ \mathsf{g}_3^{\mathsf{w}(3)}\mathsf{u} + \mathsf{q}\right] + \mathsf{t} \tag{3.32}$$

Figure 3.4 shows a factor graph that corresponds to this optimization problem for a small example setup consisting of 4 unit cells. We show points in all the Wyckoff positions (a, b, b, c, d, e, f) and show how they multiply through the lattice, generating the all the points in $\mathcal{X}$. The example optimization shown here consists of 107 points generated from only 6 points in the asymmetric unit.

Generating Symmetries in 3D

I will now jump into real world example of the sort we will encounter throughout the rest of this thesis. 3D symmetries are the most common symmetries that we encounter in urban scenes and I will provide a generative model for common 3d symmetries that we typically encounter here.

No Lattice

The Radcliffe camera building that we choose for this particular symmetry configuration is a circular building in oxford, whose symmetry can be modeled as the Axial symmetry group D_8. Similar to the Dihedral group in 2D, the generative model that corresponds to this specific instance of symmetry is shown in Figure 3.11(b). The points

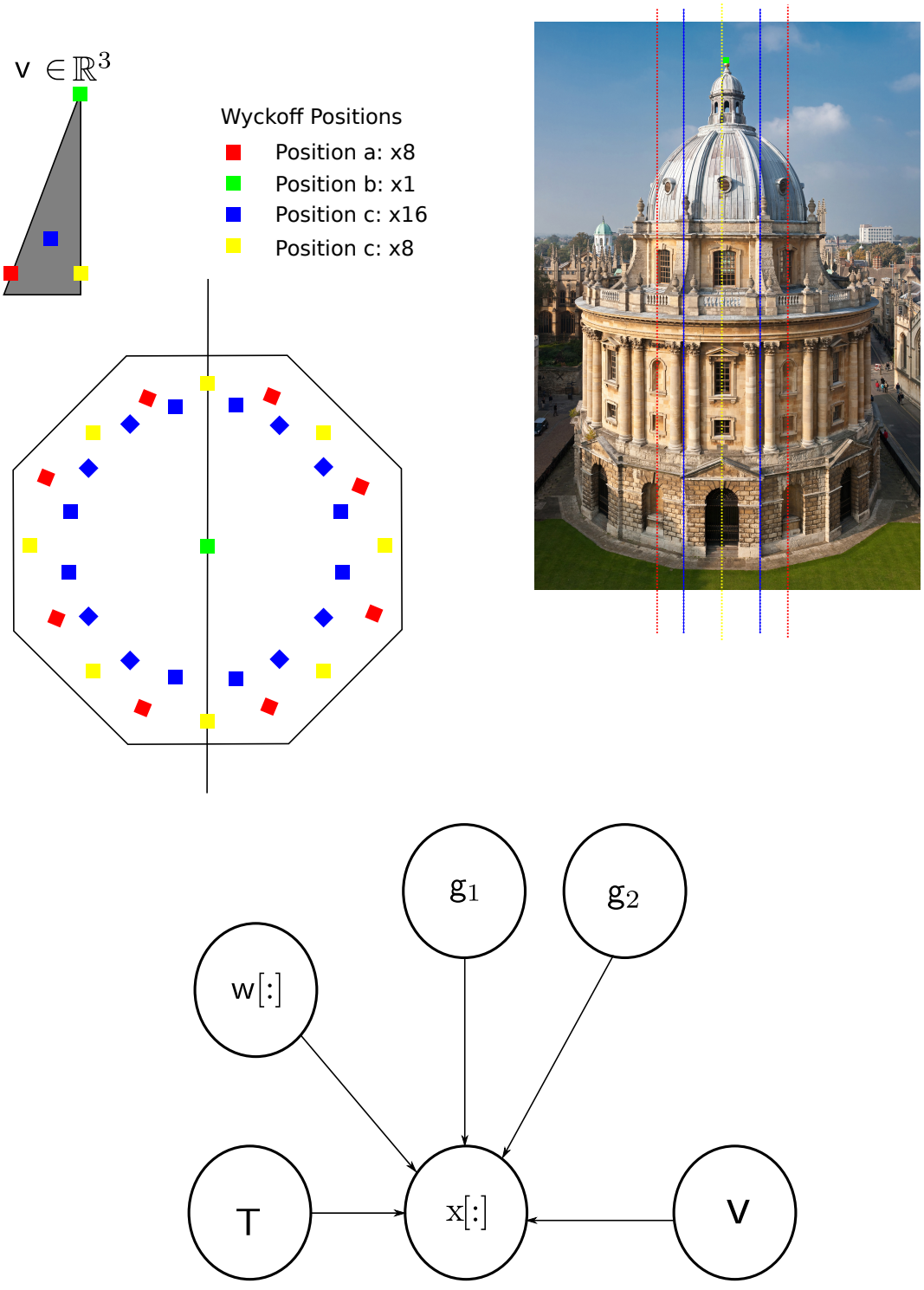


Figure 3.11: **Example of Symmetry synthesis for 1D lattice in 3D Symmetry**: Here we choose an example of the Axial symmetry group D_8. The generative model of the this group is similar to the Dihedral group D_4 , which we saw in Figure 3.5. Just like in the 2D case, there are 4 Wyckoff positions and the generators here represent 3D rotations and reflections.

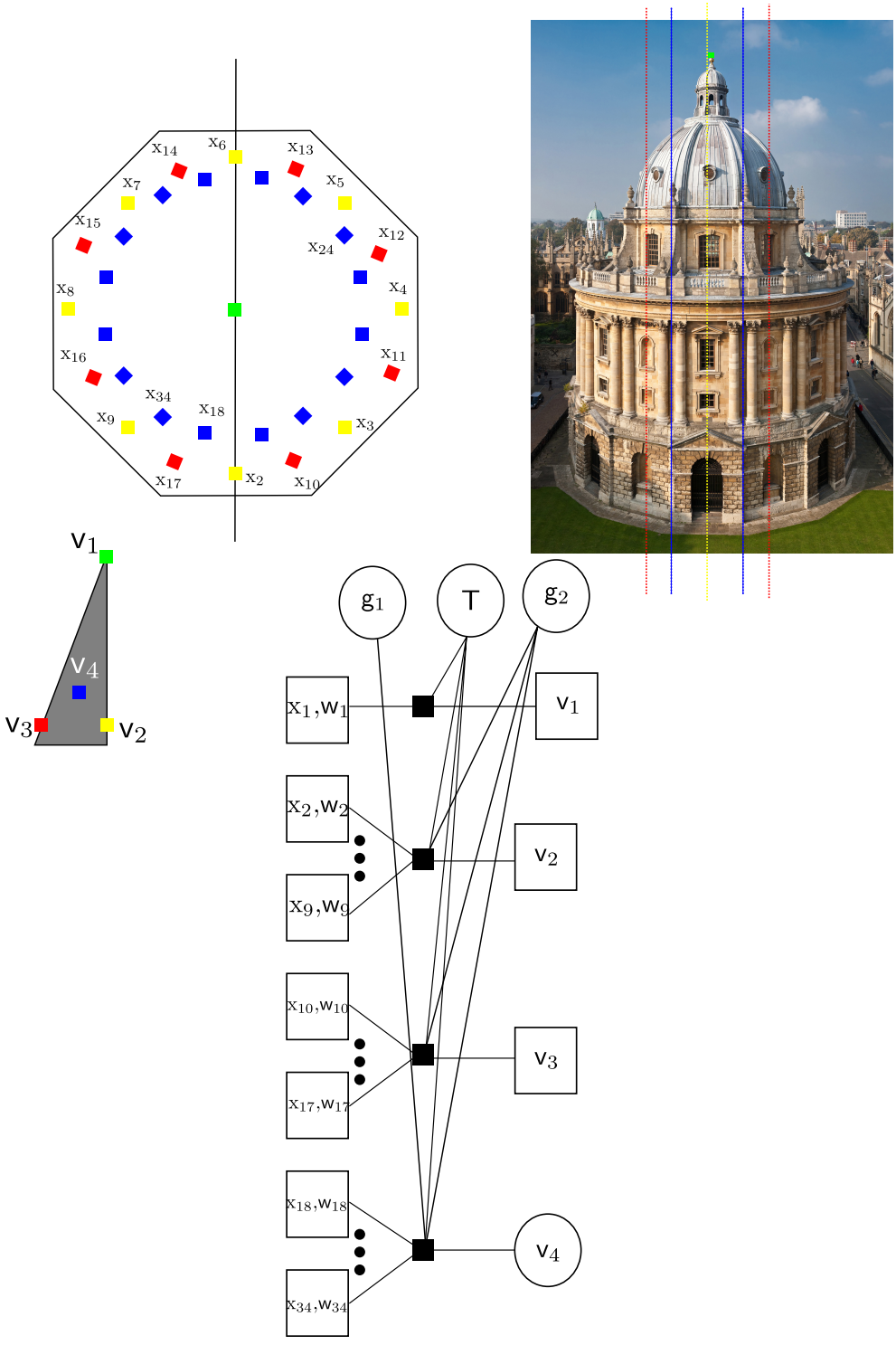


Figure 3.12: **Factor graph and example optimization for 3D symmetry with no lattice**: Factor graph for the optimization defined in Section 3.3. We are given a set of 34 measurements of point $\mathcal{X} \in \mathsf{x}_1 \dots \mathsf{x}_{34}$ and its Wyckoff indicator variables $\mathcal{W} = \{w_1 \dots w_{34}\}$ and its Miller indices $\mathcal{Q} = \{\mathsf{q}_1 \dots \mathsf{q}_{34}\}$. Our goal is to estimate the unknown location of points in the asymmetric unit $\mathcal{V} = \{\mathsf{v}_1, \mathsf{v}_2, \mathsf{v}_3, \mathsf{v}_4\}$ that generate these 34 points $\mathcal{X}$ along with The generators $\mathsf{g}_1, \mathsf{g}_2$ are the 3D analogues of reflection and rotation as described in the 2D case.

in 3D x span the space $\mathbb{R}^3$. Every point has a Wyckoff position that spans a $3-$tuple $\mathsf{w} \in \{\{a,b,c,d\},\{0,1\},\{0,1,2,3\}\}$ where the first element of the tuple specifies the Wyckoff position $\{a,b,c,d\}$ which is depicted in Figure 3.3(a), the second element specifies if the point is reflected or unreflected with respect the the y-axis and the thirst element spans the set of integers $\{1,2,3,4\}$ specifying the order of rotation. The points inside the asymmetric unit are given by $\mathcal{V}$. The 3D equivalent of these generators is given by

$$\mathsf{g}_1 = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

and the second generator is given by

$$\mathsf{g}_2 = \begin{bmatrix} 0.707 & -0.707 & 0 \\ 0.707 & 0.707 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

where g_2 corresponds to a rotation of 45^0 about the $z-$axis.

Every point x has a corresponding location inside the asymmetric unit. This location is described by the variable $\mathsf{v} = \begin{bmatrix} \mathsf{v}_x \\ \mathsf{v}_y \\ \mathsf{v}_z \end{bmatrix} \Big| \mathsf{v}_x + \mathsf{v}_y \leq 1, \sqrt{\mathsf{v}_x^2 + \mathsf{v}_y^2} \leq R, \mathsf{v}_x \geq 0$ for our choice of origin and reflection planes. In addition to this, we allow a rigid transformation of the entire symmetric structure from an arbitrarily assigned world coordinate frame we call this translation $\mathsf{T} \in \mathbb{SE}(3)$.

The joint probability distribution is similar to the 2D case, but the spaces spanned by the variable is different

$$p\left(\Theta\right) \triangleq p\left(\mathcal{X}\middle|\mathcal{V}, \mathsf{g}_1, \mathsf{g}_2, \mathcal{W}, \mathsf{T}\right) \tag{3.33}$$

where $\mathcal{X} = \left\{\mathsf{x}_j\right\}_{j=1}^{n} \middle| \mathsf{x}_j \in \mathbb{R}^3$ is the set of points that are symmetric. $\mathcal{V} = \left\{\mathsf{v}_p\right\}_{p=1}^{P} \middle| \mathsf{v}_p = \begin{bmatrix} \mathsf{v}_{px} \\ \\ \mathsf{v}_{py} \end{bmatrix} \middle| \mathsf{v}_{px} + \mathsf{v}_{py} \leq 1, \sqrt{\mathsf{v}_{px}^2 + \mathsf{v}_{py}^2} \leq R, \mathsf{v}_{px} \geq 0$ are the set of points in the asymmetric unit, $\mathcal{W} = \left\{\mathsf{w}_p\right\}_{j=1}^{n} \middle| \mathsf{w} \in \left\{\left\{a, b, c, d\right\}, \left\{0, 1\right\}, \left\{0, 1, 2, 3\right\}\right\}$ are the set of Wyckoff position for every point in $\mathcal{X}$ and $\mathsf{g}_1, \mathsf{g}_2$ are given above and $\mathsf{T} = \left(\mathsf{R}, \mathsf{t}\right) \in \mathbb{SE}\left(3\right)$. The correspondence modeling still holds from the case of 1D, i.e $\mathsf{v}_{\mathbf{p}_j} \rightarrow \mathsf{x}_j$

We can represent the generative model by means of a measurement function given by

$$\mathsf{x}_j = h_{30}\left(\mathsf{v}_p, \mathsf{g}_1, \mathsf{g}_2, \mathsf{w}_j, \mathsf{T}\right) + \eta_{30} \tag{3.34}$$

where $\eta_{30} = \mathcal{N}\left(\mathsf{x}, 0, \Sigma\right)$ is a Gaussian-corrupted noise on the point x on the symmetric structure with a covariance $\Sigma \in \mathbb{R}^{3\times3}$. The measurement function $h_{30}\left(\mathsf{v}_p, \mathsf{g}_1, \mathsf{g}_2, \mathsf{w}_j, \mathsf{T}\right)$ is

given by

$$h_{30}\left(\mathsf{v}_p, \mathsf{g}_1, \mathsf{g}_2, \mathsf{w}_j, \mathsf{T}\right) \equiv \begin{cases} \mathsf{R}\left(\mathsf{g}_1^{\mathsf{w}_j(1)} \mathsf{g}_2^{\mathsf{w}_j(2)} \mathsf{v}_{p_j}\right) + \mathsf{t} & \mathsf{w}\left(0\right) = d \\ \mathsf{R}\left(\mathsf{g}_2^{\mathsf{w}_j(2)} \begin{bmatrix} 0 \\ \mathsf{v}_{p_j y} \\ \mathsf{v}_{p_j z} \end{bmatrix}\right) + \mathsf{t},\ 0 \leq \mathsf{v}_{p_j y} \leq R & \mathsf{w}\left(0\right) = c \\ \mathsf{R}\left(\mathsf{g}_2^{\mathsf{w}_j(2)} \begin{bmatrix} \mathsf{v}_{p_j y} \\ \mathsf{v}_{p_j y} \\ \mathsf{v}_{p_j z} \end{bmatrix}\right) + \mathsf{t},\ 0 \leq \mathsf{v}_{p_j y} \leq R & \mathsf{w}\left(0\right) = b \\ \mathsf{R}\begin{bmatrix} 0 \\ 0 \\ \mathsf{v}_{p_j z} \end{bmatrix} + \mathsf{t} & \mathsf{w}\left(0\right) = a \end{cases} \tag{3.35}$$

which describes the following process, to generate a point $\mathsf{x}_j \in \mathbb{R}^2$ shown in Figure 3.6: we start from its corresponding point inside the asymmetric unit v_{p_j}. If the point is in a general position $(w\left(0\right) = d)$ (shown in blue) it has the highest multiplicity ($\times 16$) and is affected by both the generators $\mathsf{g}_1, \mathsf{g}_2$. If a point is at the center of the circle (shown in green), its location is immediately known as $\mathsf{v}_{p_j} = \left(0, 0, \mathsf{v}_{p_j z}\right)$. If a point is located along one of the two edges of the segment (shown in blue and yellow), it is only affected by the generator corresponding to rotation g_2. Its multiplicity is 8.

If we know the location of point in the general position only, Equation 3.31 reduces to

$$h_{30}\left(\mathsf{v}_p, \mathsf{g}_1, \mathsf{g}_2, \mathsf{w}_j, \mathsf{T}\right) = \mathsf{R}\left(\mathsf{g}_1^{\mathsf{w}_j(1)}\mathsf{g}_2^{\mathsf{w}_j(2)}\mathsf{v}_{p_j}\right) + \mathsf{t} \tag{3.36}$$

Figure 3.6(b) shows a factor graph that corresponds to this optimization problem for a small example setup. We have chosen the location of the points at different Wyckoff positions to show how the corresponding unknown factor would look like.

Conclusion

I conclude this chapter by summarizing how I have substantiated the second claim of my thesis

> *I provide generative models for synthesizing symmetries in 1D, 2D and 3D that encompass a larger variety of symmetries than the state of the art.*

I have provided a generative model for generating points in a symmetric structure in 1D, 2D and 3D. I have demonstrated how we can use these models to infer the variables of symmetry, from noisy observations of the symmetric structure in the form of points. In next chapters I will demonstrate how we can determine the variables of symmetry by using our generative models and observations made from a camera imaging the scene.

Chapter 4

AUTOMATIC DETERMINATION OF LATTICES IN 3D

Introduction and Literature survey

In this chapter, I aim to substantiate the third claim of my thesis:

> *I developed a method for determining the lattice parameters of those generative models from sparse and noisy SfM clouds by exploiting a novel polar space voting scheme.*

Translational symmetry is perhaps the most abundant symmetry present in urban scenes. Several man-made 3D structures exhibit some sort of translational symmetry. A lattice in an urban scene can be informally understood as the presence of an architectural element (e.g., a window, a column), which is repeated regularly within the 3D structure (a more formal definition is given in Chapter 2). We have already established that lattices form an important descriptive language of symmetries. Furthermore, they can be decouple from other forms of symmetry as our generative model demonstrates in Chapter 3. Therefore

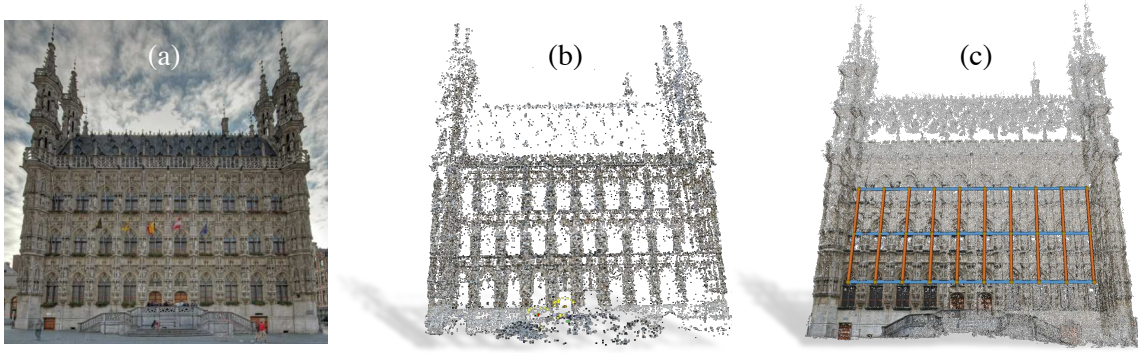


Figure 4.1: **Determining lattices using voting, an overview**: The proposed approach uses (a) a set of 2D images, and (b) a sparse 3D reconstruction to detect 2D lattice (c) Generative model of the lattice obtained from the proposed approach, for the LEUVEN dataset.

a method that can extract lattices can be beneficial in either aiding the user to look for more complex symmetries or simply on its own, provide a partial descriptive language of the scene. Moreover, it has already been demonstrated that lattices can serve an important structural constrain during structure from motion and we further bolster this understanding in Chapter 6. Detecting and describing the symmetric elements is crucial for high-level understanding of the 3D geometry (e.g, shape classification),to reduce the complexity of the corresponding model, or to edit the 3D model in a consistent manner [Mit+12]. Moreover, partial symmetries can inform Structure from Motion (SfM) and improve the quality of 3D reconstruction from an unordered set of images [Coh+12; Cey+14]

While existing literature offers well established approaches for symmetry detection from 3D models (i.e, point cloud, mesh, or volumetric grid), these techniques usually assume the model to be dense and geometrically correct. This makes their direct application to SfM challenging: point clouds reconstructed via SfM are usually noisy and sparse, and 3D symmetry detection approaches tend to perform poorly [Mit+12]. This motivates re-

cent interest towards joint solutions to detect symmetries and exploit them to improve the quality of 3D model [Coh+12; Cey+14]. While these works provide excellent contributions towards the goal of symmetry-aware SfM algorithms, they rely on strong assumptions (e.g., co planarity, availability of geometric priors [Coh+12]) or require manual annotation [Cey+14]. The algorithms in this chapter improves over related work by relaxing those assumptions and avoiding manual annotation.The key drawback in existing symmetry-based SfM approaches is that they do not allow us to recover the structure in a procedural format. This is because, with the exception of [Hon+04], lattices are used an a means to address a problem with the existing SfM pipeline rather than as a tool that can be exploited to build models that are more descriptive than sparse point clouds.[Coh+12] uses symmetries to solve the problem of drift incurred in large scale bundle adjustment problem.[Wu10] uses symmetries in a multi-view stereo setting to build dense model of the repeating element. While graphical techniques [Pau+08a; Wu+14] provide rigorous mathematical analysis for modeling and detecting and 3D symmetries from dense point clouds, there is a lack of similar development in SfM community for incorporating these models. In this chapter I seek to reduce this gap by addressing some of the issues in procedural modeling of scene from unordered image collection.

Procedural modeling of scenes from unordered image collection is a non-trivial problem because of the noisy nature of the point cloud that is typically obtained from SfM [Coh+12]. This is because procedural models typically require a $n-$fold application of a base transformation and therefore have stringent bounds on the error estimated in the base transform. Existing graphical techniques such as [Pau+08a; FS11] cannot be directly applied to sparse point clouds constructed from SfM because of the noisy and sparse nature of these clouds.

My literature review is organized according to the type of input data fed to the symmetry detection approaches. We distinguish *2D*, *3D*, and *2D-3D* approaches. 2D approaches search for repeated patterns in a single image, and can be classified into *local* and *global* approaches. Local approaches hypothesize lattice generators(a new term that we will define in this chapter) from pairwise matches and then extract dominant generators using voting schemes [LE06; PD05], greedy or exhaustive search [LM96; SZ99; Liu+05], vanishing points [WFP10], or invariance-driven techniques [GMP96]. Global approaches, instead, look for periodic texture over the entire image. Corresponding contributions include techniques based on the Fourier transform [PL94; Sun97; LCL08], Hough transform [Yip+94; CC94; TTG03], peak detection in the autocorrelation function [LWY97; LCT04; Hay+06], applications of the tiling theory [CH80], and moment-based methods [GB94]. Related work also focus on exploiting the detection of 2D symmetries for 3D reconstruction from a single image [Kan81; KK97; Hon+04; MZB95], or automatic computation of a procedural model of building facades from images [Mül+06; Wu+14].

3D approaches, the more relevant works that correspond to this chapter, search for symmetries in 3D models. Also in this case it is possible to distinguish global and local approaches. An informative overview is provided in Table 1 of [Mit+12], where techniques that focus on translational symmetries are all relevant. Global approaches use moment-based methods [Mar+06; TMS09], correlation of the extended Gaussian image [SS97], spherical harmonics [MFR04], Fourier transform [Kaz+03; FS11], and multidimensional scaling [Rav+07]. Local approaches include geometric hashing [GC06], transformation space voting [MGP06; Pau+08b], the planar reflective symmetry transform [PGR07], and graph-based approaches [Ber+08], and spectral methods [Lip+10]. For a more compre-

hensive literature review on 2D and 3D symmetry detection we refer the reader to the excellent surveys [Liu+10; Mit+12].

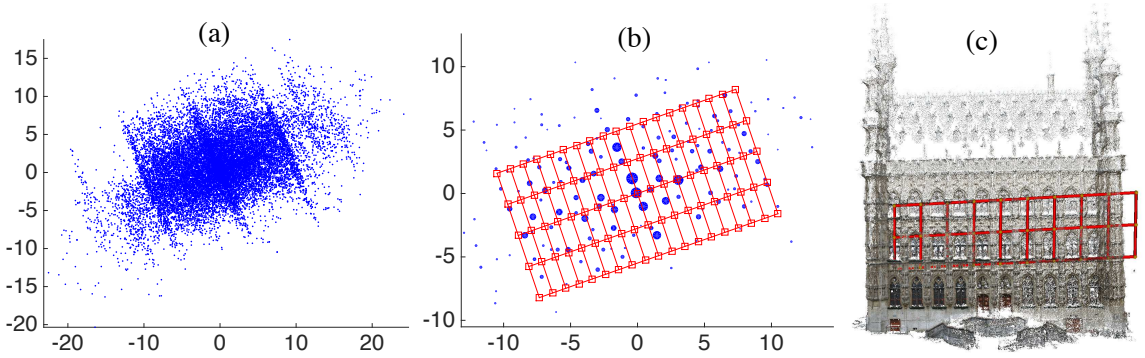


Figure 4.2: **Failure of existing methods**: LEUVEN dataset: (a) Putatives in translational transformation space. (b) Related works apply clustering, followed by grid fitting (e.g., [Pau+08a]), to infer the lattice from the grid. Grid-based detection of subdimensional lattice in SfM: (a) 3D model, (b) putative matches in transformation space, (c) clustering and grid fitting results. (c) Since the 3D model is noisy and sparse, grid fitting performs poorly.

In this chapter, I propose an approach for 2D-3D symmetry detection. This approach includes three building blocks. The first is a multi-hypothesis estimator for 3D lattice generators. We borrow key insights from [Pau+08a], which shows how to map putative symmetry transformations into 1D or 2D lattices. However, we skip grid fitting (which is unreliable on SfM data), and we show that a polar parametrization of the transformation space clearly exposes dominant lattice generators. The second block prunes the multiple hypotheses on the generators and returns the generators that are most consistent with 2D appearance. This is similar in spirit to [JTC11], while we avoid 3D surface fitting. Finally, the last block takes the estimate for the generators and jointly refines this estimate and the 3D reconstruction, will be discussed in Chapter 6.

While existing literature offers well established approaches for lattice detection from 3D models (i.e, point cloud, mesh, or volumetric grid), these techniques usually assume the model to be dense and geometrically correct. This makes their direct application to SfM challenging: point clouds reconstructed via SfM are usually noisy and sparse, and 3D symmetry detection approaches tend to perform poorly [Mit+12]. This motivates recent interest towards joint solutions to detect lattices and exploit them to improve the quality of 3D model [Cey+14]. While these works provide excellent contributions towards the goal of symmetry-aware SfM algorithms, they rely on strong assumptions (e.g., co-planarity, availability of geometric priors [Coh+12]) or require manual annotation [Cey+14]. The present paper improves over related work by relaxing those assumptions and avoiding manual annotation, while leveraging fundamental insights from our classification of symmetry.

Lattice Groups and Generators of a Lattice

The focus of this chapter is to determine the lattice of a 3D structure (if there exists one). The input is sparse(and often noisy) point cloud obtained from SfM and the output is a lattice represented by its basis B. 3D architectural elements rarely exhibit a full 3D lattice,; we are therefore often confined to detection of sub-dimensional lattice $\mathsf{B} \big| \mathsf{B} \in \mathbb{R}^{L \times L} \forall\, L < 3$. In this chapter, we specifically focus on the lattice aspect of the generative model we introduced in Chapter 2. We discover both 2D and 1D lattices in a 3D space by using the tried and tested voting scheme from the graphics community with a twist of our own design that can deal with noisy clouds. I defined lattices in Chapter 2. I reiterate its

definition here:

A **vector lattice** in $\mathbb{R}^n$ is defined by Müller *et al.* [Mül13] as follows

---

Definition 16 (10)**.** *For any basis of* $\mathbb{R}^n$*, the subgroup of all linear combinations with integer coefficients of basis vectors* b *forms a vector lattice.*

Since the lattice represents the discrete translational symmetry generators of the symmetry group $\mathcal{G}_S$ of a set of points, $\mathcal{X}$, there is a system of generators assosiated with it. In order to understand the system of generators of a lattices, we need to first understand what we mean by **lattice group** G_L. Conway *et al.* [CS13] defines a lattice in $\mathbb{R}^n$ from a group-theoretic point of view as:

Definition 17. *A lattice in* $\mathbb{R}^n$ *is the symmetry group of discrete translational symmetry in* n *directions.*

The system of generators of the lattice group G_L is a set of n translational vectors that corresponds to the basis vectors of the lattice. If the lattice has a translation and rotation relative to arbitrary global coordinate, the basis vector is affected by the rotational component of the pose with the translation having no effect on the basis. Much of the next section is dedicated to establishing this fact, a reader who is interested in the voting can skip to the next section.

A particular model of symmetry as defined by our generative model, uniquely determines lattice but the converse is not true. Let us briefly revisit the definition of lattices a group of

the discrete translational symmetry and how this viewpoint modifies the generative model described in Section 6.3. Every point in $\mathcal{X}$ is generated from its corresponding point in the asymmetric unit $(\mathcal{U}, \mathcal{V})$ by a two stage process **(1)** by first transforming the point by combination of the system of generators $g_p = \{{}_p\mathsf{g}_1 \ldots {}_p\,\mathsf{g}_P\}$ that define the symmetry operations within the unit cell and (2) translating the transformed point by a vector specified by the basis of the lattice B, its pose in 3D space ${}_w\mathsf{T}_a$, and the Miller indices $\mathcal{Q}$.

In this section we establish the relation between the lattice of dimension L located in a space of dimension N specified by the tuple $(\mathsf{B}, {}_w\mathsf{T}_a)$ and its generator system $g_l = \{{}_l\mathsf{g}_1 \ldots {}_l\,\mathsf{g}_L\}$. Consider two N dimensional points $\mathsf{x}_1, \mathsf{x}_2 \in \mathbb{R}^N$ in a L dimensional lattice defined by a basis B and a transformation ${}_w\mathsf{T}_a$ with respect to a global coordinate frame, having Miller indices of q_1 and q_2 that correspond to the **same** point in the unit cell $(\mathsf{u}\prime, \mathsf{v}\prime)$. The relative location between $\mathsf{x}_1, \mathsf{x}_2$, specified by a transformation ${}_1\mathsf{T}_2$ can be obtained using the definition of a point in the sub-dimensional lattice in Chapter 2 as

$$\begin{aligned} \mathsf{x}_1 &= {}_w\mathsf{R}_l \begin{bmatrix} \mathsf{B} \cdot (\mathsf{u}\prime + \mathsf{q}_1) \\ \mathsf{v}\prime \end{bmatrix} +_w \mathsf{t}_l \\ \mathsf{x}_2 &= {}_w\mathsf{R}_l \begin{bmatrix} \mathsf{B} \cdot (\mathsf{u}\prime + \mathsf{q}_2) \\ \mathsf{v}\prime \end{bmatrix} +_w \mathsf{t}_l \end{aligned} \tag{4.1}$$

The vector that connects the points x_1 and x_2 is obtained by taking their difference as

$$\mathsf{x}_1 - \mathsf{x}_2 =_w \mathsf{R}_l \begin{bmatrix} \mathsf{B} \cdot (\mathsf{q}) \\ 0 \end{bmatrix} \tag{4.2}$$

where $\mathsf{q} = \mathsf{q}_1 - \mathsf{q}_2 \in \mathbb{Z}^L$. Consider the same problem above where a L dimensional lattices is specified by its generator system $g_l = \{{}_l\mathsf{g}_1 \ldots {}_l\,\mathsf{g}_L\}$. The vector connecting the points x_1 to x_2 can specified as

$$\mathsf{x}_1 - \mathsf{x}_2 = {}_l\mathsf{g}_1^{(\mathsf{q}_1(0)-\mathsf{q}_2(0))} \ldots {}_l\mathsf{g}_L^{(\mathsf{q}_1(L-1)-\mathsf{q}_2(L-1))} \tag{4.3}$$

If the representation of the generators are in the form of translation vectors, ${}_l\mathsf{g}_1 \in \mathbb{R}^N$, the composition operation takes the form of a sum operator and the above equation becomes

$$\mathsf{x}_1 - \mathsf{x}_2 = \sum_{l=0}^{L-1} \mathsf{q}_1\,(l) - \mathsf{q}_2\,(l)\,{}_l\mathsf{g}_1 \tag{4.4}$$

If we assume that the variable q is the same in the both the Equations 4.4 and 4.2, then we get the following relationship

$${}_w\mathsf{R}_l\mathsf{B} = [{}_l\mathsf{g}_1 \ldots {}_l\mathsf{g}_L 0] \tag{4.5}$$

Since $|{}_w\mathsf{R}_l| = 1$, we can see that, given as set of generators ${}_l\mathsf{g}_1 \ldots {}_l\mathsf{g}_L$, we can obtain the matrix ${}_w\mathsf{R}_l$ and Busing RQ factorization of the augmented matrix given in Equation 4.5

$${}_w\mathsf{R}_l\mathsf{B} = \mathbf{RQ}\,([{}_l\mathsf{g}_1 \ldots {}_l\mathsf{g}_L 0])$$

which leads to a rotation matrix ${}_w\mathsf{R}_l$ and an upper triangular matrix B.

Circular Lattices

Circular Lattices

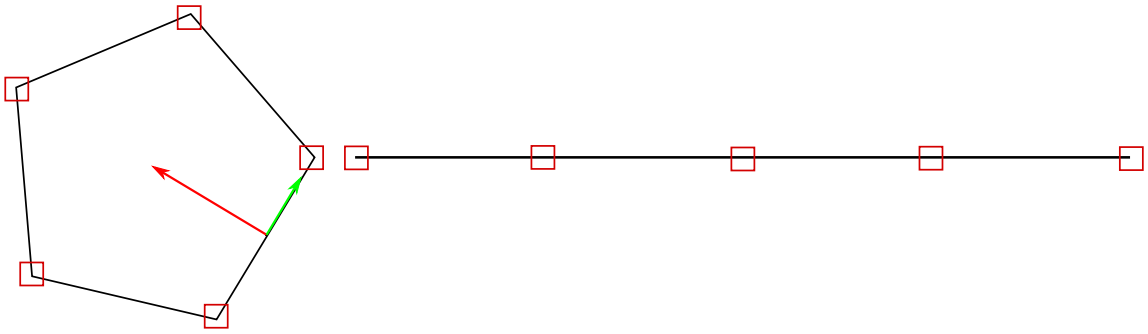


Figure 4.3: **[Circular Lattices]:** Circular Lattices can be seen as a translational symmetry along the circumference of the object on which concepts like unit unit cells and basis can be defined. Related works have addressed this problem and many related ones including cases of skewed rotations (Lee *et al.* [LL10]).

While cyclic groups as shown in Figure 4.3 do not have a vector lattice as described in Definition 16, we can still define the concept of a unit cell as follows.

Consider the scenario shown in Figure 4.3: There are n points in an object that exhibits a cyclic symmetry C_L that is to be determined. In this case, $\mathsf{q} \in \mathbb{R}$ is the number of times by which a rotation of $\mathsf{B} \in \mathbb{SO}(2)$ has to be applied to bring the a point in the unit cell $\mathsf{u}\prime_p$ into coincidence with its corresponding point in the circularly symmetry structure C_L. We use the same notation so that a reader can draw a $1:1$ map between the translational case and the rotational case. The unit cell is defined as a pose ${}_{=}\mathsf{T}_{(\cdot,\mathsf{R_t})} \in \mathbb{SE}(2)$where t is the location of the center of the circle. Geometrically, in the 2D case, the unit cell

is a circular arc and the identity rotation $_{=}\mathsf{R}_I 2$ is that rotation that brings the $x-$axis to coincide with the lower arm of the segment. Similar to the above scenario, as input, we are given as set of measurements $\mathcal{Z} = \{\mathsf{z}_k \in \mathbb{R}^2\}_{k=1}^{K}$where the subscript k represents the index of the measurement . Also, the correspondence information is given by two functions $J\Big|\{0\dots K-1\} \rightarrow \{0\dots n-1\}$ and $Q\Big|\{0\dots K-1\} \rightarrow \mathcal{Q}$ where J specifies the j^{th} point in the unit cell that corresponds to the k^{th} measurement and Q specifies the Miller index corresponding to the k^{th} measurement. The measurement function $h_{2D}^{C}\left(,\mathsf{T}_\mathsf{B},\mathsf{u}\prime_{\mathsf{j}_k},\mathsf{v}\prime_{\mathsf{j}_k};\mathsf{q}_k\right)$ predicts the location of a feature z_k as given in Equation 2.7 and the measurement function for the circular case is given by

$$h_{2D}^{C}\left(,\mathsf{T}_\mathsf{B},\mathsf{u}\prime_{\mathsf{j}_k},\mathsf{v}\prime_{\mathsf{j}_k};\mathsf{q}_k\right) = f\mathsf{R}\left(\mathsf{B},\begin{bmatrix}\mathsf{u}\prime_{\mathsf{j}_k}\\ \mathsf{v}\prime_{\mathsf{j}_k}\end{bmatrix},\mathsf{q}\right)+\mathsf{t} \tag{4.6}$$

where $f\left(\mathsf{B},\begin{bmatrix}\mathsf{u}\prime_{\mathsf{j}_k}\\ \mathsf{v}\prime_{\mathsf{j}_k}\end{bmatrix},\mathsf{q}\right)$ is a function that rotates the point in the unit cell $\mathsf{u}\prime_{\mathsf{j}_k}$ q times by the rotation B, If B is specified by a 2×2 orthogonal rotation matrix, this can be written as

$$f\left(\mathsf{B},\mathsf{u}\prime_{\mathsf{j}_k},\mathsf{q}_k\right) = \mathsf{B}^{\mathsf{q}}\cdot\begin{bmatrix}\mathsf{u}\prime_{\mathsf{j}_k}\\ \mathsf{v}\prime_{\mathsf{j}_k}\end{bmatrix} \tag{4.7}$$

where B^{q} is the matrix for the rotation R raised to the power $\mathsf{q}\in\mathbb{R}$.

Problem definition

Feature Extraction and Transformation Space

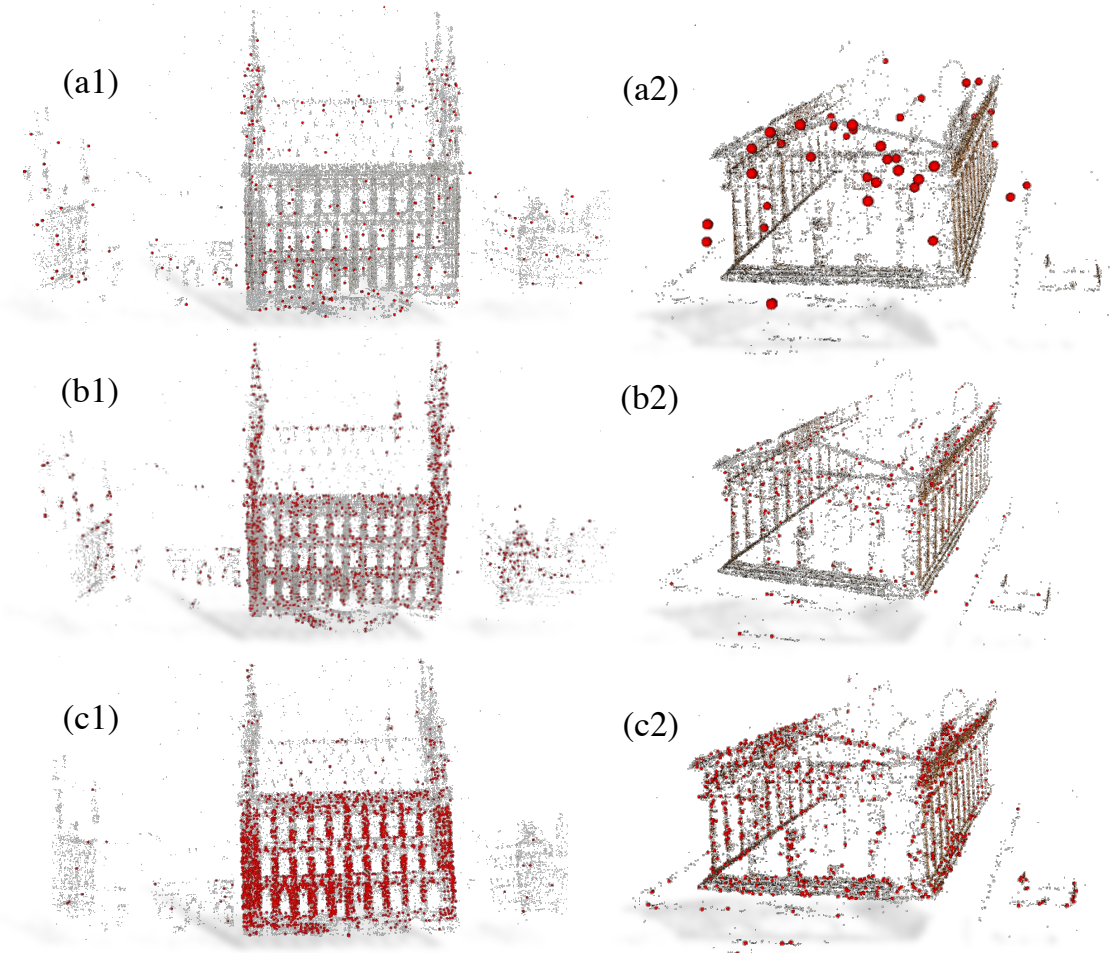


Figure 4.4: **3D Sift Keypoint detection**: 3D sift keypoint detection for the LEUVEN dataset (left column) and for the NEPTUNE dataset (right column). The number of detected keypoints depends on the search radius. The figure shows (by rows) the detected keypoints for increasing values of the search radius r_{SIFT}:$(a1, a2)$ r_{SIFT}= 10^{-2}, $(b1, b2)$ $r_{SIFT} = 2.5 \cdot 10^{-2}$, $(c1, c2)$ $r_{SIFT} = 10^{-1}$. For the results shown in this document and in the main paper, we use $r_{SIFT} = 2.5 \cdot 10^{-2}$.

This section describes how to create a set of putative transformations for exposing the lattice of a 3D scene and represent them in a suitable transformation space. We first detect 3D features on the point cloud $\mathcal{X}$. While related works obtains these features by uniform sampling [Pau+08a], we use 3D SIFT keypoints, and obtain n points $\mathsf{x}_{s1} \dots \mathsf{x}_{sn} \in \mathbb{R}^3$ (an example is given in Figure 4.4). Figures 4.1 and 4.4 (b) show the 3D features in red, for the LEUVEN dataset. For each feature x_{si}, we associate a set of neighboring points $\mathcal{P}_i$, which are within a ball of radius $\bar{r}_s$ from x_{si}. Then, for each pair (i, j) of features, we compute a relative transformation by applying ICP to the patches $\mathcal{P}_i$ and $\mathcal{P}_j$. This gives a set of putative transformations $\mathcal{T} \doteq \{T_{ij} : i, j = 1 \dots n\}$. The intuition is that some of the detected features are arranged according to a lattice geometry, and its parameters should emerge from the pairwise relations T_{ij}. Each T_{ij} describes a rigid transformation and can be written as $T_{ij} \doteq (\mathsf{R}_{ij}, \mathsf{t}_{ij})$ where, $\mathsf{R}_{ij} \in \mathbb{SO}(3)$(a 3D rotation)and $\mathsf{t}_{ij} \in \mathbb{R}^3$(a 3D translation). Our goal is to use the set $\mathcal{T}$ to find generators $\mathsf{g}_1, \dots, \mathsf{g}_k$ corresponding to a K-dimensional lattice in a 3D model. The difficulty in doing this stems from the fact that the set $\mathcal{T}$ contains many outliers pairs (i, j) that are not related by a lattice relation and, even for the pairs that belong to the lattice, the transformation T_{ij} is produced by a composition of an unknown number of unknown generators $\mathsf{g}_1, \dots, \mathsf{g}_k$ of the lattice, since the lattice dimension is not specified beforehand. To make the problem more tractable and since we are mainly interested in symmetries in urban architecture, we focus on the following lattices in 3D:

- 1D lattice in 3D symmetry : This has one generator g_1.
- 2D lattice in 3D symmetry: This has two generators $\mathsf{g}_1, \mathsf{g}_2$.

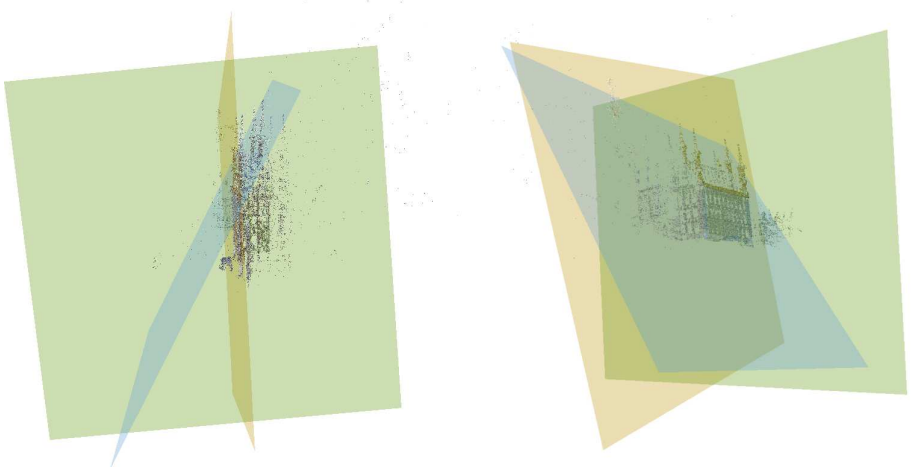


Figure 4.5: [**RANSAC**]: The 2D plane obtained using RANSAC in the transformation space visualized in 3D.

- 2D Circular lattice in 3D symmetry: This has two generator $\mathsf{g}_1, \mathsf{g}_2$.

We want to identify the symmetries listed above in noisy point clouds. As in [Pau+08a], we partition the set of putative transformations $\mathcal{T}$ into two subsets: the subset $\mathcal{T}_\mathsf{t}$ of transformations having almost null rotation, and the subset $\mathcal{T}_\mathsf{R}$ of transformations having a significant rotation component (i.e., rotation angle greater than a threshold ϵ_R). More formally, $\mathcal{T}_\mathsf{t} \doteq \{T_{ij} = (\mathsf{R}_{ij}, \mathsf{t}_{ij}) \in \mathbb{SE}(3) : \|\mathsf{R}_{ij} - I_3\|_F < \epsilon_\mathsf{R}$, where $\|\cdot\|_F$ is the Frobenius norm, I_3 is the 3×3 identity matrix, and ϵ_R is a threshold; $\mathcal{T}_\mathsf{t}$ is then $\mathcal{T}_\mathsf{R} \doteq \mathcal{T} \setminus \mathcal{T}_t$.

The putatives in $\mathcal{T}_\mathsf{t}$ describe pure translations, hence we can visualize them as points in $\mathbb{R}^3$. This set has to contain any evidence of translational symmetries. The seminal work [Pau+08a] shows that, when considering 1 and 2-parameter lattices, the translations defining the magnitude and direction of the lattices lie in a subspace of $\mathbb{R}^3$ (intuitively, the 2 generators of a 2-D lattice only span a plane in 3D space). Therefore, according to [Pau+08a] we re-parametrize the (translational) transformation space as a

2D space. This re-parametrization requires to fit a plane to the 3D points in $\mathcal{T}_{\mathsf{t}}$ (using RANSAC), and then projecting the resulting inliers to the estimated plane [Pau+08a]. Figure 7 in [Pau+08a] provides an intuitive description of this process. We take the same approach and use RANSAC to fit a plane in the 3D transformation space and project the inliers to this plane, such that and we write each putative transformation as a 2D vector $\mathsf{x}_{ij} \in \mathbb{R}^2$ [Pau+08a]. Figure 4.5 shows the plane that is obtained from the transformation space visualized on the 3D point cloud. We can see that these correspond to one the the three dominant sub dimensional lattices of the 3D structure. We denote the corresponding set of putatives $\mathcal{T}_{\mathsf{t}}^{2d} \doteq \{\mathsf{x}_{ij}\} \left| \mathsf{x}_{ij} \in \mathbb{R}^2 \right.$. The set of these putatives, $\mathcal{T}_{\mathsf{t}}^{2d} \doteq \{\mathsf{x}_{ij} \in \mathbb{R}^2\}$, is called the *translational transformation space*. An example of $\mathcal{T}^{2d}$ is given in Figure 4.2{a} for the LEUVEN dataset.

Similarly, the set $\mathcal{T}_{\mathsf{R}}$ has to contain evidence of rotational symmetry. A mild assumption is that the axis of the rotational symmetry is vertical (this is most commonly the vertical axis in urban architecture). Hence we select the subset of $\mathcal{T}_{\mathsf{R}}$ having rotation axis close to the vertical, and we store the corresponding rotation angles. This allows us to re-parametrize the putatives as a set of rotation angles $\phi_{ij} \in \{-\pi, \pi\}$. We call $\mathcal{T}_{\mathsf{R}}^{1d} \doteq \phi_{ij}$ the *rotational transformation space*.

While the creation of the transformation space leverages the elegant formulation of [Pau+08a], we follow a different approach to "explore" this transformation space, searching for the symmetry generators (Section 4.6). To motivate our approach, we now briefly discuss issues arising from the direct application of grid fitting [Pau+08a] to estimate the generators from the SfM reconstruction. Pauly [Pau+08a] propose to first perform clustering, to expose recurrent transformations; the result of clustering for the LEUVEN dataset is shown

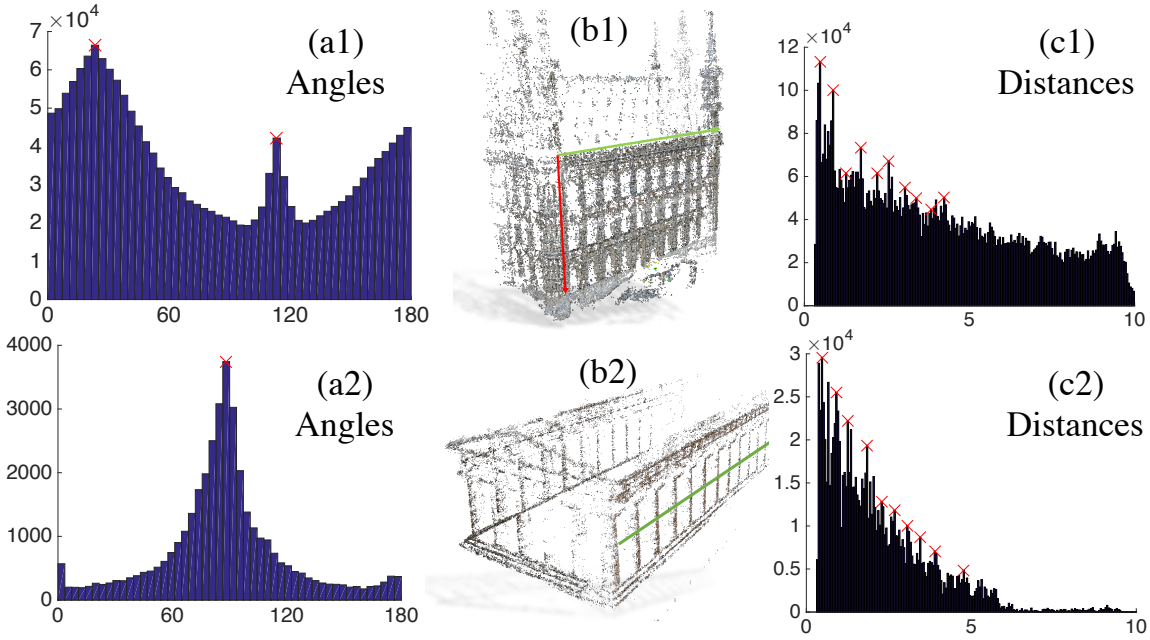


Figure 4.6: **Angle Histograms to Detect Symmetries**: Histogram of the angles θ_{ij} in the polar transformation space for the LEUVEN (a1) and for the NEPTUNE (a2) datasets. Corresponding 3D directions are visualized in (b1) and (b2), respectively. (c1,c2) Histograms of the distances ρ_{ij} for the points in the transformation space having angles θ_{ij} close to the highest peaks in (a1) and (b1), respectively.

in Figure 4.2(b) in blue. Then, they fit a regular lattice using nonlinear optimization. The issue with this approach is that the result of clustering is very cluttered when dealing with SfM data; moreover, the optimization problem underlying grid fitting is nonconvex and has many local minima, hence it can easily converge to an incorrect estimate of the generators (Figure 4.2(c)). This issues have been also acknowledged in [Cey+14]. In the following section we propose an alternative approach that avoids clustering and achieve grid fitting by re-parameterizing the transformation space in a way that clearly exposes dominant symmetry directions. We first show that the translation directions can be easily computed when parameterizing the transformation space in polar coordinates.

Polar Voting Scheme

Let us start our discussion with the translational transformation space $\mathcal{T}^{2d}$ (the rotational space $\mathcal{T}_{\mathsf{R}}^{1d}$ will be discussed at the end of this section). The key insight of our approach is the following: the presence of the repetitive structure like a lattice in the 3D model implies that many of the transformations $\mathsf{x}_{ij} \in \mathcal{T}^{2d}$ will be (approximately) in the form $\mathsf{x}_{ij} = n \cdot \mathsf{g}_k$ ($k \in \{1, 2\}$) i.e., will be produced by an $n-$fold repetition along the generator $\mathsf{g}_k \in \mathbb{R}^2$ (this is also one of the motivations for the grid fitting of Figure 4.2(b)). Moreover, the $2-$vector g_k can be written as $\mathsf{g}_k = \delta_k \cdot \mathbf{u}_k$, where $\delta_k \in \mathbb{R}$ is the *repetition period* and the unit vector $\mathbf{u}_k \in \mathbb{R}^2$ $\|\mathbf{u}_k\| = 1$ is the *repetition direction*. Therefore, all the putatives generated by g_k become $\mathsf{x}_{ij} = (n\delta_k) \cdot \mathbf{u}_k$, i.e., they share the same direction, while they possible have different norms(depending on the unknown number of repetitions n). We argue that the generator direction $\mathbf{u}_i$ is easy to compute, while the difficult part of the problem is to estimate the period δ_i, because of the ambiguity in the number of repetitions n.

In order to exploit this insight, we propose to re-parametrize the transformation space $\mathcal{T}^{2d}$ in polar coordinates, i.e., we write each $\mathsf{x}_{ij} \in \mathcal{T}^{2d}$ as an angle/distance pair (θ_{ij}, ρ_{ij}). We call the resulting set of pairs (θ_{ij}, ρ_{ij}) the *polar* transformation space (PTS). All the putatives corresponding to repetitions along the generator g_k, which we wrote in Cartesian coordinates as $(n\delta_k) \cdot \mathbf{u}_k$, can be expressed in polar coordinates as $(\theta_k, n\delta_k)$, where θ_k is the angle between the direction $\mathbf{u}_k$ and the horizontal axis; these putatives exhibit the same angle θ_k,(this is essentially the repetition direction $\mathbf{u}_k$), but possibly different distances(as the distance is influenced by the number of repetitions n). Therefore, an histogram plot

of the θ_{ij} of the putatives should reveal generator directions. Therefore, we can expose the dominant symmetry directions from an histogram plot of the angles $\{\theta_{ij}\}$; an example for the LEUVEN dataset is given in Figure 4.6(a1). The peaks correspond to directions that occur in many putative pairs, and these are the most likely to capture a generator direction. The histogram in Figure 4.6(a1) is obtained from the raw transformations in $\mathcal{T}^{2d}$ without preprocessing (we do not apply any type of clustering or filtering). The histogram clearly exposes the dominant repetition directions, visualized in 3D The 3D directions corresponding to the peaks of the histogram are shown in Figure 4.6(b1). Note that the polar histogram also gives a clear picture of the *number* of directions remember that we are looking for 1 and 2-D lattices, hence (we do not know a-priori if the repetitions are along 1 or 2 directions). For instance, for the NEPTUNE dataset (Figure 4.6(b2)), the lateral facade of the temple only contains a 1-D lattice, and this is correctly captured by the histogram of Figure 4.6(a2), which has a single prominent peak. In summary, from the angle histogram we can infer the number of translational generators (this L for an L-dimensional lattice) and the directions u_k $(k = 1, \ldots, K)$, which can be computed from the peaks (if the peak occurs at an angle θ_k, then $u_k = [\cos(\theta_k)\ \sin(\theta_k)]\, T$.

Now, to completely characterize the generators $\mathsf{g}_k = \delta_k \cdot \mathbf{u}_k$, we need to compute the period $\delta_k \forall k = 1, \ldots, K$. We argue that this is the difficult part of the symmetry detection problem: even the putatives corresponding to the lattice provide a guess on δ_k only up to an unknown integer. (recall that these putatives can be written as $\mathsf{x}_{ij} = (n\, \delta_k) \cdot \mathbf{u}_k$). We propose to use our polar representation also for the computation of the period. For each dominant direction $\mathbf{u}_k$ (or, equivalently, θ_k), we select the points that are along this direction (in polar coordinates, these are the points $((\theta_{ij},\ \rho_{ij})$ with $|\theta_{ij} - \theta_k| \leq \epsilon_\theta$, where

ϵ_θ is a threshold); then we compute the histogram of the distances ρ_{ij} associated to these points. Ideally, this histogram should have peaks at integer multiples of the period δ_k.(this is essentially the underlying idea in order to compute the period associated to the generator direction). Recalling that each putative in the PTS is written as $\mathsf{x}_{ij} = (\theta_{ij}, \rho_{ij})$, we propose to select the putatives along each symmetry directions u_k, and consider only the distance from the origin ρ_{ij} for each of this inliers. Then, the histogram of the distances ρ_{ij} exposes the period of the repetition. This histogram does not exhibits clear dominant peaks. Examples of distance histograms are given in Figure 4.6(c1), and in Figure. Unfortunately, the distance histograms are not as clean as the angular ones (Figure 4.6(a1)(a2)). This is due to both the presence of outliers and the fact that a single generator creates multiple peaks, spaced at regular intervals ($1-$fold repetitions, $2-$fold, etc.), "diluting" the density. In Figure 4.6 we plot the distance histogram along the direction of the horizontal generator of the LEUVEN dataset. The red crosses in Figure 4.6(c1)(c2) denote the 10 highest peaks, selecting the peaks that have distance at least $0.3m$. The selection of the peaks can be done using the standard Matlab function "findpeaks". From this selection, we remove the peak at the origin, which is always present, but uninformative (the putatives at the origin correspond to transformations that map each patch $\mathcal{P}_i$ to itself). From our experience, selecting the largest peak [LCT04] can lead to erroneous period estimate . While we could attempt fitting a 1D lattice to match the peaks in Figure 4.6(c1)(c2), we prefer to return multiple hypotheses for the periods, corresponding to the m largest peaks. Then we use appearance information to prune incorrect period hypotheses (Section 4.7) (while the 1D lattice would be a purely geometric check).

In summary, the proposed PTS allows computing the number of generators K, the direc-

tion of each generator u_k, and m hypothesis for the repetition period $\delta_k^1, \ldots, \delta_k^m$. Hence we build m hypotheses for each generator as $\mathsf{g}_k^j = \delta_k^j \cdot u_k$, $j = 1, \ldots, m$. We argue that the advantage of our approach over grid fitting is that we identify a subproblem that is easy to solve (estimating K and the symmetry directions), and a subproblem that is hard (estimating the period), but can be dealt with independently for each direction and reduces to a 1D problem. This is in contrast to grid fitting which attempts to estimate all the generators (i.e., solves a hard problem in larger dimension), and works on the clusters of Figure 4.2(b) which can be a poor description of the transformation space.} Indeed, we argue that the generator direction is easy to compute, while the difficult part of the problem is to estimate the period δ_i, because of the ambiguity in the number of repetitions.

We conclude this section by commenting on the detection of **circular lattices**. In Section 4.5 we discussed how to obtain the 1D rotational transformation space $\mathcal{T}_{\mathsf{R}}^{1d} = \{\phi_{ij}\}$, which only contains 1D putatives. Similar to the distance histogram of Figure_4.6, the histogram of the angles in $\mathcal{T}_{\mathsf{R}}^{1d}$, e.g., the one in Figure 4.7(c), has multiple peaks spaced at regular intervals, corresponding to (e.g., for the structure in Figure 4.7(a) we have peaks at 30^0, 60^0, etc. produced by the $n-$fold rotational repetitions. Our approach retains the highest m peaks, corresponding to m hypotheses for the rotational symmetry, which are then pruned as discussed in Section 4.7.

Appearance based generator ranking

As discussed in Section 4.6, our approach uses the geometry of the 3D model (encoded in the putative transformations) to produce multiple hypotheses for the generators of the

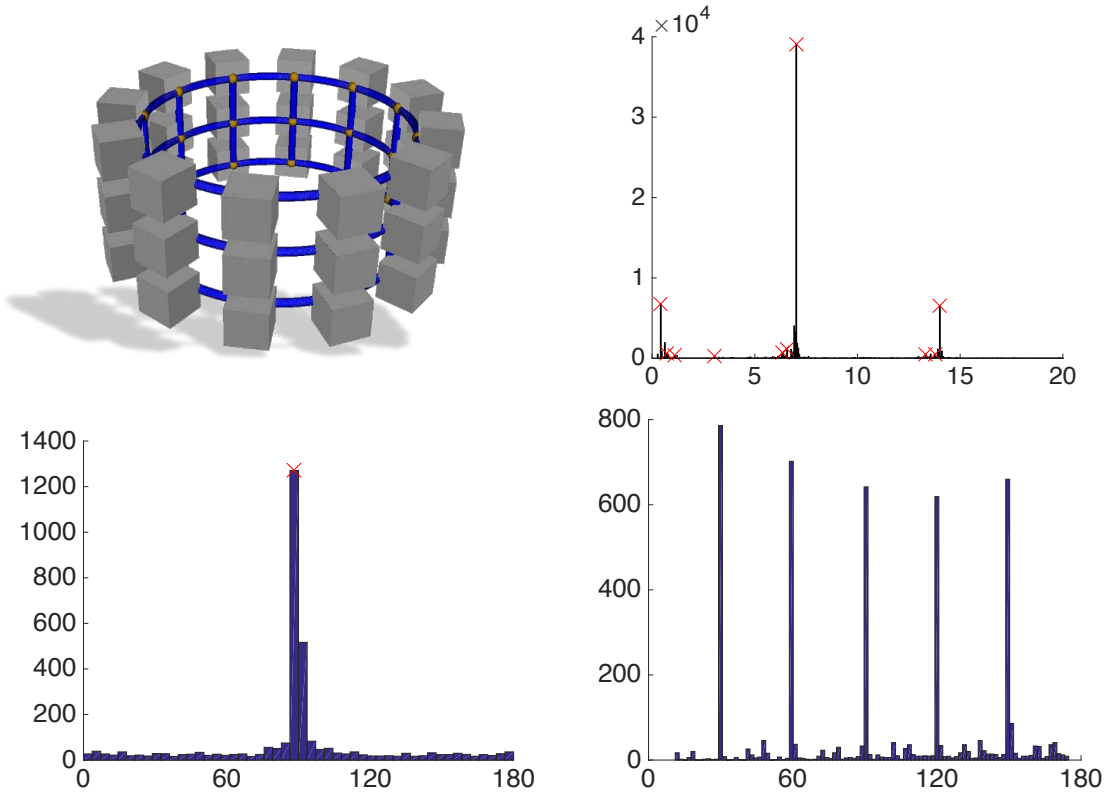
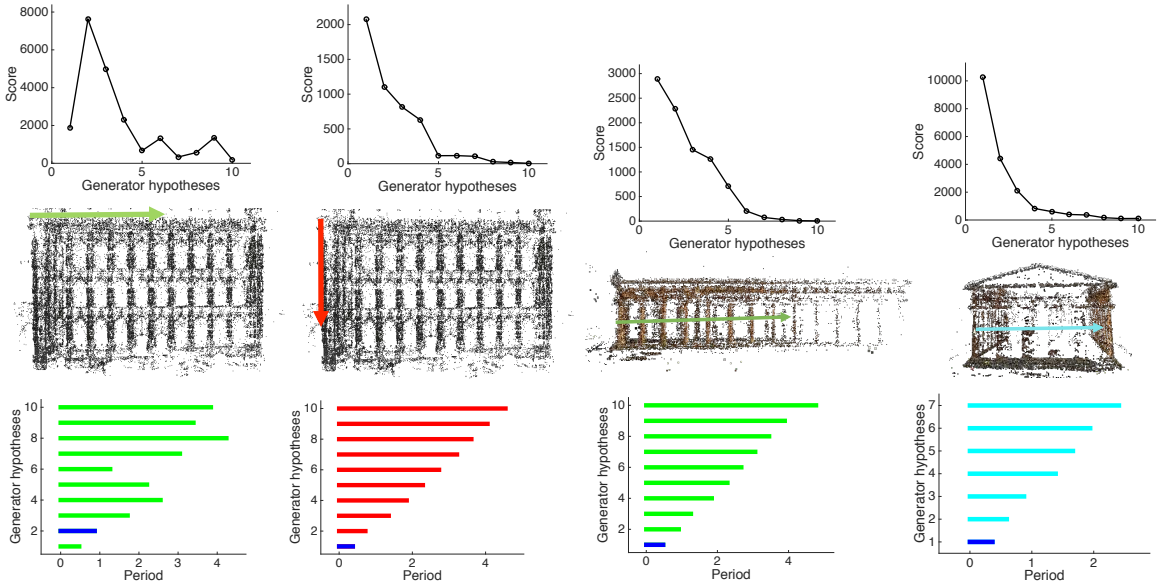


Figure 4.7: **Determination of Circular Lattices using Voting**: (a) Simulated dataset with 2D Circular Lattice. (b) Histogram of angles in the PTS having direction aligned with the vertical. (c) Histogram of angles $\{\phi_{ij}\}$ in the rotational transformation space $\mathcal{T}_{\mathsf{R}}^{2d}$.



(a) LEUVEN Ranking (b) NEPTUNE ranking

Figure 4.8: **Appearance based Generator Ranking**:

lattice. In this section we propose to prune these hypotheses using appearance information from the images. Our appearance-based verification step is similar to the one used in [JTC11]. The intuition is that a good generator is one that maps a patch to another one having similar appearance (this is essentially the concept of lattice). Since each point in the 3D model is obtained via SfM, it has a corresponding feature descriptor in each image in which it was seen. Therefore, the appearance check reduces to verifying that the generator maps a point to another point that has a similar descriptor.

We use this insight to devise a generator ranking scheme. For each generator hypothesis g_k^i we do the following. We consider each point x_0 in the 3D model and we call $\tilde{d}0$ the corresponding descriptor. We apply the transformation encoded in the generator g_k^j and get a second point x_1(this is the result of a rotation or translation of x_a). Then we look in a

ball of radius $\bar{r}_a$ around x_1 and we check all the descriptors $\tilde{d}1$ corresponding to points in this ball. If at least one $\tilde{d}1$ is close enough to the original descriptor $\tilde{d}0$ (angle between the descriptors smaller than a threshold $\bar{\tilde{d}}$), then we say that x_0 "agrees" with the generator g_k^j. Repeating this process for each point in the cloud we have a score for g_k^j, which is simply the number of points that agrees with the generator. Therefore, we select the best generator as the one having highest score. Scores for the hypotheses in the LEUVEN and NEPTUNE datasets are reported in Figure 4.8a and Figure 4.8b. We remark that considering a ball around the transformed point $\point_1$ allows us to mitigate the impact of to better cope with noise in the generator estimates as well as the sparsity of the point cloud.

Lattice Identification: The first step is to discover $n-$fold repetitions, for each of generator g_k; This is fairly simple and shares the same idea as the appearance-based verification of Section 4.7. Let us consider the current estimate of the generator, say g_i(this procedure has to be repeated for each generator). We consider each point x_0 in the 3D model and we call $\tilde{d}0$ the corresponding descriptor. Then we apply the transformation encoded in the generator g_k and get a second point x_1. Inside a ball of radius $\bar{r}$ around x_1, we look for the point having descriptor which is closest to $\tilde{d}0$. If the closest descriptor is within a distance $\bar{d}$ from $\tilde{d}0$, then we establish a 1-fold symmetry relation between x_0 and x_1, encoded by the tuple $(\mathsf{x}_0, \mathsf{x}_1, \mathsf{g}_k, 1)$ (the 1 stands for $1-$fold); we store the tuple in a set $\mathcal{S}$, called the *symmetry relations set*. We repeat the same procedure for x_1, trying to establish another $1-$fold relation with a third point x_2. If this succeeds we add the tuple $(\mathsf{x}_0, \mathsf{x}_2, \mathsf{g}_k, 2)$ to $\mathcal{S}$. We iterate this procedure until no more repetition satisfies the check on the descriptor. As a result, we populate a set of symmetry relations $\mathcal{S}$, which describes the discovered $n-$fold repetitions.

Conclusion

I conclude this chapter by summarizing the claim that this chapter addressed. I have demonstrated that voting in a polar transformation space can successfully extract 1D and 2D lattice from a noisy reconstructed point cloud in 3D. I have further shown that we can achieve this without the need for any manual intervention. I have defined a related idea called circular lattices and demonstrated how we can treat this as a special case of translational lattice and thereby infer the generators of a circular lattice, which is, in reality an Axial symmetry group in 3D. In the next chapter we will see how we can obtain the asymmetric unit and the full parameters of the symmetry model defined in Chapter 2 using only image(s) and a single manual intervention stage.

Chapter 5

SINGLE VIEW RECONSTRUCTION UNDER SYMMETRY

In this chapter, I aim to substantiate the fourth claim of my thesis:

> *Given a single view and a known symmetry type, I provide a semi-automatic method for determining the 3D structure of the scene as comprised of non-decomposable elements known as asymmetric units.*

In the previous chapter, I demonstrated a technique for extracting a subset of the generative model described in Chapter 3 from 3D point clouds obtained from SfM of many images. This subset pertains to the lattice parameters of the generative model. I was able to achieve this by decoupling the generators corresponding to the lattice, as elements of discrete translational symmetry group. The advantage of this approach was that it was completely automatic while the disadvantage was that, we are restricted to the identification of lattices only. In addition, we depend on an initial reconstruction of the scene which is not always possible in the presence of symmetry.

In this chapter we address the problem of single view reconstruction and estimation of variables of the full generative model that we described in Chapter 3 using only a single

image as input along with a single user interaction. I provide an algorithm for describing the 3D structure of a symmetric scene using the full generative model which requires us to establish the parameters of the point groups of symmetry in addition to the lattice parameters we saw in the previous chapter. I make the following assumptions about the input : (1) a user can specify an interesting symmetry element with a single manual intervention (2) the symmetry group of the scene is known beforehand.

State-of-the art works that perform structure estimation from a single image exhibiting symmetries can only deal with simplistic symmetry models such as the fundamental symmetry operation that we described in Chapter 2. The idea of using symmetry to determine the 3D structure from a single image is not new. Several authors have addressed this problem; some of the newer works which talk about single image reconstruction exclusively in the context of symmetry are [Jia+15; KZP11]. Some authors have explored the topic of extracting the lattices from a single image [LLH04]. Other related works that focus on the topic of single image reconstruction from symmetry are [Kan81; KK97; Hon+04; MZB95]. These works are restrictive to the type of symmetry; for example, Koser *et al.* [KZP11] provide results on bilateral symmetries, and Hong *et al.* show results on translational symmetries, while disregarding other symmetries. Another closely related topic is determination of procedural model of building facades from images [Mül+06; Wu+14].In addition to making simplistic assumptions about symmetries, the goal of these works is to establish the scene depth alone with little or no interest in determining the elemental units that make up symmetry, with the exception of procedural modeling, where the converse of that statement is true.

Related works that **detect** 3D symmetry from single image are numerous and we pick the

following ones which have an element of interest that is particularly close to what we are trying to achieve [TK12; Fun+17; TFA15; LSD13; Par+09; LL13; Coh+17; WCL16]; Here we will briefly discuss those that are closest to my works by focusing on those that determine scene geometry along with symmetry. These works fall into the detection of either (1) rotational and reflectional symmetry or (2) translational symmetry (lattice). While standalone reflection and rotational symmetry detection can be incredibly useful especially to determine the type of model to use particularly if the model has some symmetry operations that contain these two aspects, we focus here on the works that discover lattices. This is because lattices while defined in the strictest crystallographic sense mainly address translational symmetry, a deformed lattice can be used to detect rotational symmetry as well. However, in this work I do not exploit the full power of symmetry detection algorithms although doing this will incorporate a degree of automaticness into the algorithm. I plan to incorporate this into a future work.

Our method addresses the drawbacks in the state of the art technique by providing a single view reconstruction technique that can utilize the full power of the generative model of symmetry to identify and exploit a range of complex symmetries and lattices detailed in Chapters 2 and 3. In addition to providing a dense reconstruction of the scene, our method also determines the building blocks that define the generative model of symmetry, the **asymmetric units**. This attempts to combine the elements of procedural modeling and depth estimation in the techniques described above. This is a natural consequence of the rigorous modeling of symmetry that we showed in Chapter 2.

We validate our approach on a variety of existing datasets that exhibit the complex symmetries, group Pm [Coh+12] and on rotational symmetry dataset of a dihedral group

D_n where n is unknown. To my knowledge there is no work that provides a full generative model of 3D symmetries from a single image, while modeling the subtle variations in depth in 3D. In particular, I make the following 3 contributions in this Chapter: (1) a technique for obtaining the 3D structure of the scene upto scale with a single image (2) a generative description expressing the scene using elemental building blocks called asymmetric units (3) a camera calibration technique that uses any naturally occurring symmetric scene as a virtual calibration grid to establish its intrinsic and extrinsic parameters.

I approach this chapter by first defining the complexity assosiated with single view reconstruction by describing an optimization problem. I then approach this chapter similar to the way I have progressed through the other chapters; dividing the observed symmetry into categories and using a generative model that pertains to that particular classification to define an optimization problem for jointly establishing symmetries and the 3D structure while extracting the asymmetric unit. In the last section I show how we can use an initial estimate to densify the obtained points to provide a reconstruction that rivals SfM results.

Single Image Reconstruction

I first define the problem of structure estimation from a single view without symmetry before proceeding to the problem of structure estimation under symmetry. Consider a set of n 3D points $\mathcal{X}$ being observed by a camera c. A 3D point $\mathsf{x}_j \in \mathcal{X}$ generates a measurement $\mathsf{z}_k \in \mathcal{Z}$. There is a total of K measurements, $\mathcal{Z} = \{\mathsf{z}_k\}_{k=0}^{K-1}$ where $K \leq n$ because some points may not generate valid measurements either because of occlusions or because they are outside the camera's field of view. We model the two-way correspondence information

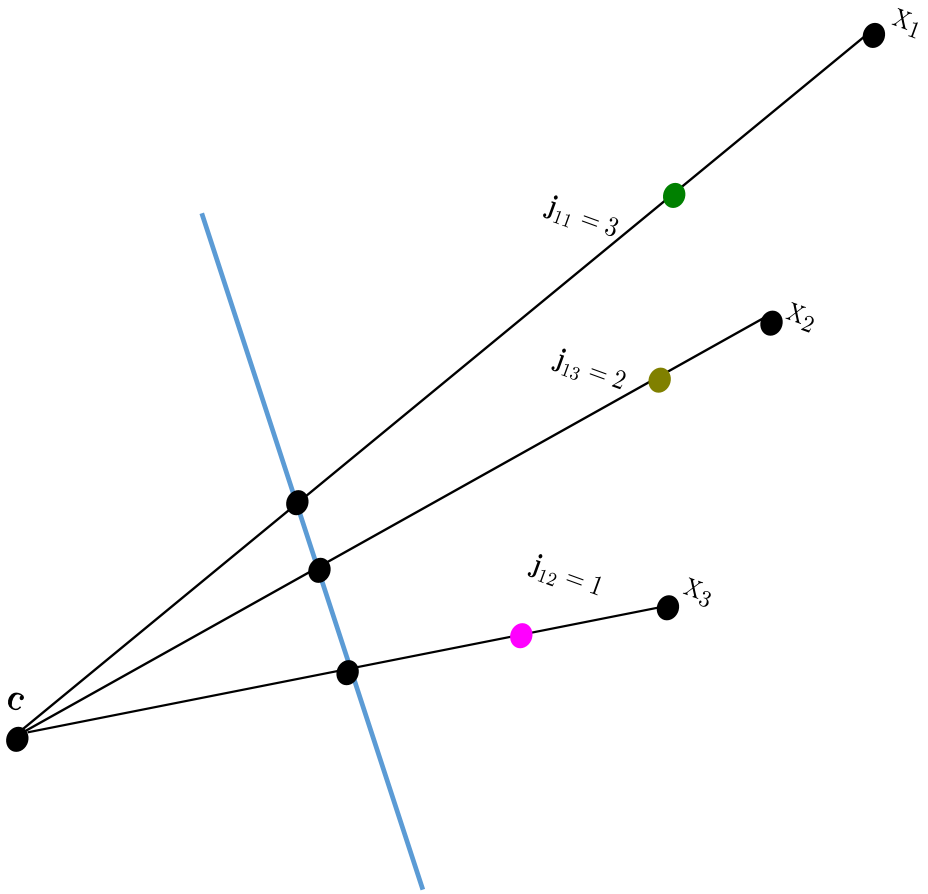


Figure 5.1: **Ambiguity in Single view reconstruction**: Every point can be at a different depth and yet generate the same measurement in the image. The points $\mathcal{X} = \{\mathsf{x}_1, \mathsf{x}_2, \mathsf{x}_3\}$ generate three measurements on the image plane of camera c. We can freely move the location of points along the ray and we would still get the exact same measurement in the image. This is the ambiguity assosiated with reconstruction from a single view. However, we we are able to establish additional relationship between points $\mathsf{x}_1, \mathsf{x}_2, \mathsf{x}_3$, we can eliminate this ambiguity. For example if we know that $\mathsf{x}_1, \mathsf{x}_2, \mathsf{x}_3$ are at the same depth from the camera, the points can no longer move along the ray independent of each other, but can still move together such that they collectively maintain the same depth from the camera. This is the planarity constraint and it reduces the DOF of the system by 2 (3 to 1).

between a 3D point x_j and a measurement z_k using the correspondence variable $\mathbf{j}$ where $\boldsymbol{j}_k$ determines j^{th} index of the element in $\mathcal{X}$, given the k^{th} measurement in $\mathcal{Z}$, therefore the 3D point $\mathsf{x}_{\boldsymbol{j}_k}$ generates the measurement z_k. Similarly, $\mathbf{k_j}$ determines the indices of the measurement given the index of the 3D point j, therefore a measurement z_k is being generated by a 3D point $\mathsf{x}_{\mathbf{k_j}}$.

In order to optimize for the unknown 3D points and the cameras, we need to specify the measurement function. The measurement function $h\left(c, \mathsf{x}_{\mathbf{k_j}}\right) \mathbb{R}^3 \rightarrow \mathbb{R}^2$ predicts the measurement z_k as

$$\mathsf{z}_k = h\left(c, \mathsf{x}_j\right) + \eta_{\mathsf{z}} \tag{5.1}$$

where η is the measurement noise and $j = \boldsymbol{j}_k$. The most common measurement function used in computer vision is obtained by transforming the 3D point to local camera frame of the c followed by a projection

$$h\left(c, \mathsf{x}_j\right) = \Pi\left({}_w\mathsf{R}_c^T\left(\mathsf{x}_j - {}_w\,\mathsf{t}_c\right)\right) \tag{5.2}$$

where the pose of the camera is given by ${}_w\mathsf{T}_c = \left({}_w\mathsf{R}_c, {}_w\,\mathsf{t}_c\right) \in \mathbb{SE}\left(3\right)$ and $\Pi\left(\cdot\right)$ is the projection operator that depends on the camera calibration model.

Now that we have defined our measurement function we can now define the log likelihood which will serve as the energy function whose global maxima corresponds to the optimal value of the unknown camera c and 3D points $\mathcal{X}$. The maximum likelihood estimate $\left(\mathcal{X}^\star, c^\star\right)$ is given by

$$\left(\mathcal{X}^\star, c^\star\right) = \arg\max_{\left(\mathcal{X}, c\right)} \mathcal{L}\left(\mathcal{X}, c; \mathcal{Z}, J\right) \tag{5.3}$$

$$-\log \mathcal{L}\left(\mathcal{X}, c; \mathcal{Z}, J\right) = \frac{1}{2\sigma^2} \sum_{k=1}^{K} \left\| \mathsf{z}_k - h\left(c, \mathsf{x}_{j_k}\right) \right\|^2 \tag{5.4}$$

where J is the known correspondence information between 3D points and measurement.

The above energy function (Equation 5.4) has many solutions and location of the 3D points $\mathcal{X}$ and camera c **cannot be uniquely determined.** This is because, every 3D point $\mathsf{x}_j \in \mathcal{X}$ that needs to be estimated provides 3 degrees of freedom while only generating 2 constraints (x and y direction in the image). This leads to an under-constrained linear system where the number of degrees of freedom is always greater than the number of unknowns. This can either result in no solution or an infinite set of solutions neither of which is useful for us.

If we know the relationship between some of the 3D points, we can increase the number of constraints in Equation 5.4 by modeling this relationship between those points and in some cases it is possible to obtain the 3D location of these points. For example, consider the following case: we divide the input set of points into two equal groups $\mathcal{X} = \{\mathcal{X}_a, \mathcal{X}_b\}$ and know that for any point $\mathsf{x}_{a_j} \in \mathcal{X}_a$ there is a unique corresponding point in $\mathsf{x}_{b_j} \in \mathcal{X}_b$ such that $\mathsf{x}_{a_j} = f\left(\mathsf{x}_{b_j}, \Theta\right)$ where Θ are the function parameters of any non-linear function $f\left(\cdot\right)$. In this case, we only need to estimate the location of point set $\mathcal{X}_b$ and the parameters Θ of the function $f\left(\cdot\right)$, to fully determine the set of all points $\mathcal{X}$.

$$-\log \mathcal{L}\left(\mathcal{X}_b, c, \Theta; \mathcal{Z}, J\right) = \frac{1}{2\sigma^2} \sum_{k=1}^{K_A} \left\| \mathsf{z}_k - h\left(c,\ f\left(\mathsf{x}_{b_j}, \Theta\right)\right) \right\|^2 + \frac{1}{2\sigma^2} \sum_{k=1}^{K_B} \left\| \mathsf{z}_k - h\left(c,\ \mathsf{x}_{b_j}\right) \right\|^2 \tag{5.5}$$

where K_B is the number of measurements generated by directly imaging the set of points in $\mathcal{X}_b$ and K_A is the number of measurements obtained by imaging the set of points $\mathcal{X}_A$.

The variable J now models the relationship between the measurement $\mathcal{Z}$ and its corresponding point in $\mathcal{X}_b$. In this case, we notice that every point $\mathsf{x}_{b_j} \in \mathcal{X}_b$, now generates two measurements: one by the virtue of being directly projected onto the camera and the other being virtually projected through the function $f(\cdot)$. We therefore have 4 constraints with only 3DOF added to the system. However, the parameters Θ add additional DOFs which will again make the system under constrained once again. However, we can overcome this issue now by simply observing more measurements since every new measurement adds an additional constraint until the number of constraints is greater than the DOF of Θ. In the next 2 sections we will see two examples 3D symmetries that provide additional constrains that allow us to estimate the structure of the scene from a single image.

Single Image Reconstruction with No Lattices

Problem Definition

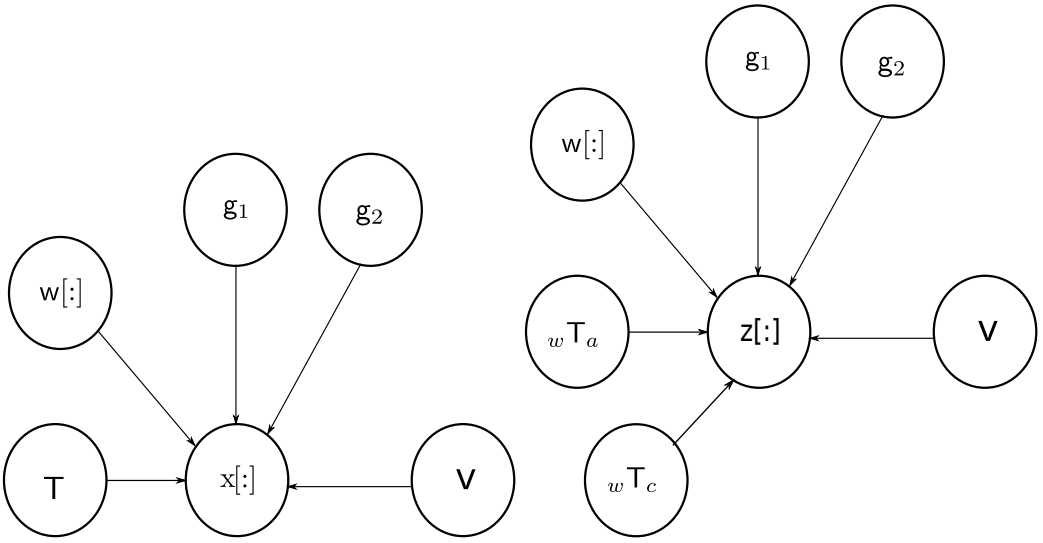
In this section, I utilize the generative model for the Dihedral symmetry group D_8 that we introduced in Chapter 3 and extend it by adding the variables pertaining to modeling the camera geometry.

In this section we will describe the generative model of a Dihedral group D_N where N is unknown. A majority of the equations are similar to D_8 with the differences occurring in the Wyckoff variable $\mathcal{W}$ and the generator pertaining to rotational symmetry g_2

Figure 5.2: **Example of Complex 3D symmetry with No lattice**: The Royal Albert Hall belongs to the dihedral group D_N. By the classification scheme described in Chapter 2, this symmetry belongs to the 3D symmetry with no lattice. In this chapter will demonstrate how we can reconstruct this building by using a single image of the building and a single user interaction.

The measurement prediction function is given by

$$\mathsf{x}_j = h_{30}\left(\mathsf{v}_p, \mathsf{g}_1, \mathsf{g}_2, \mathsf{w}_j, \mathsf{T}\right) + \eta_{30} \tag{5.6}$$



(a) Generative Model for 3D point x_j (b) Generative Model for measurement z_k

Figure 5.3: **Generative Model for the Dihedral Symmetry group** (D_n)**, a 3D symmetry with No lattice**: This is the symmetry group exhibited by the Royal Albert Hall building: (a) Bayes Net showing the generative model for a symmetric 3D point (b) Updated Model taking into consideration the camera parameters.

where $h_{30}(\cdot)$ is given by

$$h_{30}(\cdot) \equiv \begin{cases} {}_w\mathsf{R}_a \left(\mathsf{g}_1^{\mathsf{w}_j(1)} \mathsf{g}_2^{\mathsf{w}_j(2)} \mathsf{v}_{p_j}\right) +_w \mathsf{t}_a & \mathsf{w}(0) = d \\ {}_w\mathsf{R}_a \left(\mathsf{g}_2^{\mathsf{w}_j(2)} \begin{bmatrix} 0 \\ \mathsf{v}_{p_j y} \\ \mathsf{v}_{p_j z} \end{bmatrix} \right) +_w \mathsf{t}_a,\ \mathsf{v}_{p_j y} \leq R & \mathsf{w}(0) = c \\ {}_w\mathsf{R}_a \left(\mathsf{g}_2^{\mathsf{w}_j(2)} \begin{bmatrix} \mathsf{v}_{p_j y} \cos\left(\frac{360}{2N}\right) \\ \mathsf{v}_{p_j y} \sin\left(\frac{360}{2N}\right) \\ \mathsf{v}_{p_j z} \end{bmatrix} \right) +_w \mathsf{t}_a,\ \mathsf{v}_{p_j y} \leq R & \mathsf{w}(0) = b \\ {}_w\mathsf{R}_a \begin{bmatrix} 0 \\ 0 \\ \mathsf{v}_{p_j z} \end{bmatrix} +_w \mathsf{t}_a & \mathsf{w}(0) = a \end{cases} \tag{5.7}$$

where

$$\mathsf{g}_1 = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5.8}$$

and

$$\mathsf{g}_2 = \begin{bmatrix} \cos\left(\frac{360}{N}\right) & -\sin\left(\frac{360}{N}\right) & 0 \\ \sin\left(\frac{360}{N}\right) & \cos\left(\frac{360}{N}\right) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5.9}$$

$\mathcal{W} = \{\mathsf{w}_p\}_{j=1}^{n} \Big| \mathsf{w} \in \{\{a,b,c,d\},\{0,1\},\{0,1,2,N-1\}\}$ is the Wyckoff indicator variable, $({}_w\mathsf{R}_a, {}_w\,\mathsf{t}_a) \in \mathbb{SE}(3)$ is the pose of the unit cell with respect to the world frame, $\mathsf{v} = \begin{bmatrix} \mathsf{v}_x \\ \mathsf{v}_y \\ \mathsf{v}_z \end{bmatrix} \Big| \mathsf{v}_x + \mathsf{v}_y \leq 1, \sqrt{\mathsf{v}_x^2 + \mathsf{v}_y^2} \leq R, \mathsf{v}_x \geq 0$ is the location of the point in the asymmetric unit.

We now model the camera projection of the point x_j using Equation 5.2 which leads to the following measurement prediction function for a measurement z_k being generated by

a point in the asymmetric unit v_p

$$\mathsf{z}_k = \Pi \begin{cases} {}_c\mathsf{R}_w^{\mathsf{T}}\ {}_a\mathsf{R}_w \left(\mathsf{g}_1^{\mathsf{w}_j(1)} \mathsf{g}_2^{\mathsf{w}_j(2)} \mathsf{v}_{\mathbf{p}_{\mathbf{j}_k}} \right) + {}_c\mathsf{R}_w^{\mathsf{T}}\, \mathsf{t}_a - {}_w\,\mathsf{t}_c & \mathsf{w}(0) = d \\ {}_c\mathsf{R}_w^{\mathsf{T}}\ {}_a\mathsf{R}_w \left(\mathsf{g}_2^{\mathsf{w}_j(2)} \begin{bmatrix} 0 \\ \mathsf{v}_{\mathbf{p}_{\mathbf{j}_k} y} \\ \mathsf{v}_{\mathbf{p}_{\mathbf{j}_k} z} \end{bmatrix} \right) + {}_c\mathsf{R}_w^{\mathsf{T}}\, \mathsf{t}_a - {}_w\,\mathsf{t}_c,\ \mathsf{v}_{\mathbf{p}_{\mathbf{j}_k} y} \leq R & \mathsf{w}(0) = c \\ {}_c\mathsf{R}_w^{\mathsf{T}}\ {}_a\mathsf{R}_w \left(\mathsf{g}_2^{\mathsf{w}_j(2)} \begin{bmatrix} \mathsf{v}_{\mathbf{p}_{\mathbf{j}_k} y} \cos\left(\frac{360}{2N}\right) \\ \mathsf{v}_{\mathbf{p}_{\mathbf{j}_k} y} \sin\left(\frac{360}{2N}\right) \\ \mathsf{v}_{\mathbf{p}_{\mathbf{j}_k} z} \end{bmatrix} \right) + {}_c\mathsf{R}_w^{\mathsf{T}}\, \mathsf{t}_a - {}_w\,\mathsf{t}_c,\ \mathsf{v}_{p_j y} \leq R & \mathsf{w}(0) = b \\ {}_c\mathsf{R}_w^{\mathsf{T}}\ {}_a\mathsf{R}_w \begin{bmatrix} 0 \\ 0 \\ \mathsf{v}_{\mathbf{p}_{\mathbf{j}_k} z} \end{bmatrix} + {}_c\mathsf{R}_w^{\mathsf{T}}\, \mathsf{t}_a - {}_w\,\mathsf{t}_c & \mathsf{w}(0) = a \end{cases} \tag{5.10}$$

where Π is the camera projection function and $({}_w\mathsf{R}_c, {}_w\,\mathsf{t}_c) \in \mathbb{SE}(3)$ is the camera pose. Additionally we need to model the correspondence relation between the measurement index k, the index of the 3D point j and the index of the point in the asymmetric unit p. While we introduced the variable J in the previous section. We use the index p to represent the p^{th} point in the asymmetric unit. Similar to the previous case, $\mathbf{p}_j = p$, where $\mathbf{p}$ is map from the p^{th} point in the asymmetric unit to the j^{th} point x_j. Therefore, we can move back and forth between a measurement $\mathsf{z}_k \in \mathbb{R}^2$ projected from a point $\mathsf{x}_j \in \mathbb{R}^3$ and a point in the asymmetric unit v_p as follows: Given z_k, it is a projection of point $\mathsf{x}_{\mathbf{j}_k}$ which is being generated by the point in the asymmetric unit $\mathsf{v}_{\mathbf{p}_{\mathbf{j}_k}}$. The relationship between all the points

in the asymmetric unit $\mathcal{V}$ and the points $\mathcal{X}$ is collectively modeled by the variable P.

At the time of writing this thesis, we do not have a method for identifying the special position, we therefore use only the general position as the model for predicting a measurement

$$\mathsf{z}_k = \Pi\left(\left({}_w\mathsf{R}_c\right)^{\mathsf{T}}\ {}_w\mathsf{R}_a\left(\mathsf{g}_1^{\mathsf{w}_j(1)}\mathsf{g}_2^{\mathsf{w}_j(2)}\mathsf{v}_{\mathbf{p}_{\mathbf{j}_k}}\right) + \left({}_w\mathsf{R}_c\right)_w^{\mathsf{T}}\mathsf{t}_a - {}_w\,\mathsf{t}_c\right) \tag{5.11}$$

Using the measurement function described above we can now optimize for the symmetry parameters $\Theta = \{N, {}_w\,\mathsf{R}_a, {}_w\,\mathsf{t}_a\}$, the points in the asymmetric unit $\mathcal{V}$ and the camera $c = ({}_w\mathsf{R}_c, {}_w\,\mathsf{t}_c, K)$ where K is the calibration matrix that determines the projection function $\Pi(\cdot)$. The optimal value of these parameters is a maximum likelihood estimate of $c, \mathcal{V}, \Theta$

$$(c^\star, \mathcal{V}^\star, \Theta^\star) = \max_{(c,\mathcal{V},\Theta)} \mathcal{L}\left(c, \mathcal{V}, \Theta; \mathcal{Z}, J, \mathsf{P}, \mathcal{W}\right) \tag{5.12}$$

where P, J models the correspondence information between measurements $\mathcal{Z}$ and points in asymmetric unit $\mathcal{V}$ and $\mathcal{W}$ models the Wyckoff position of every 3D point that is generated and has a valid measurement in $\mathcal{Z}$. The negative log likelihood is given by

$$-\log\left(\mathcal{L}\left(c, \mathcal{V}, \Theta; \mathcal{Z}, J, \mathsf{P}, \mathcal{W}\right)\right) = \frac{1}{2\sigma^2}\sum_{k=1}^{K}\left\|\mathsf{z}_k - \mathsf{z}^\dagger{}_k\right\|^2 \tag{5.13}$$

where $\mathsf{z}^\dagger{}_k$ is the predicted measurement using Equation 5.10 and z_k is the actual measurement that is corrupted by a Gaussian noise $\mathcal{N}(0, \sigma I_2)$. In the next section we will show how we can reduce the number of free parameters for this optimization by analyzing the Gauge freedom of the system.

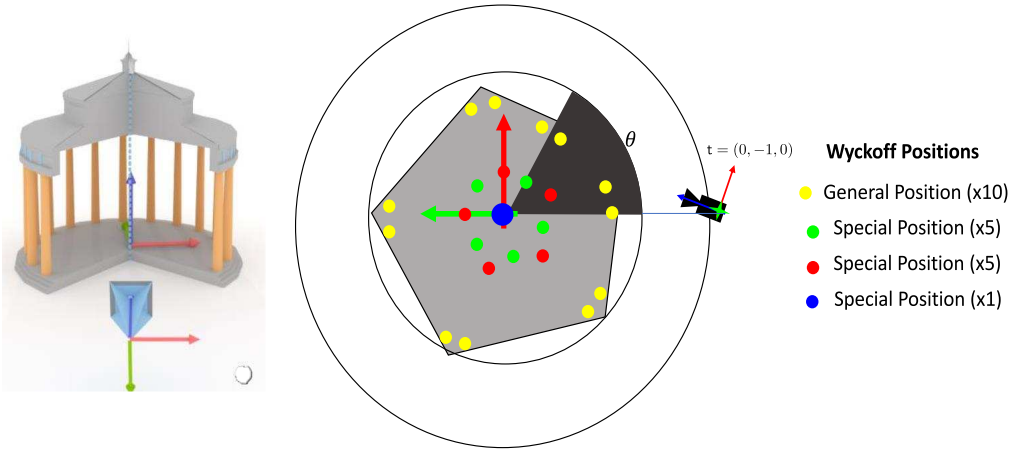


Figure 5.4: **Gauge Freedom for camera imaging a scene that exhibits Dihedral symmetry (D_N):** A camera imaging a Dihedral group contains 4 Wyckoff positions out of which three are in the special position (point on the axis of rotation(blue) and points on the planes of reflection(green) and on the plane bisecting them (red) and one is in the general position(yellow)). The choice of coordinate system inside the asymmetric unit determines the number of degrees of freedom of the system. In Section 5.2.2, we discuss how this configuration eliminates redundant DOF in the system. The net DOF of the system is given by $5 + K + 3 * |\mathcal{V}|$ where K is the number of DOF of the intrinsics of the camera and $|\mathcal{V}|$ is the number of points in the asymmetric unit.

Gauge Freedom

The variables in Equation 5.10 is an under-constrained system as there are several redundant degrees of freedom (DOF). This is a consequence of allowing a unique choice of coordinate frames of reference for representing asymmetric units $\mathcal{V}$, generators g and camera c. In this section, we will eliminate the redundant degrees of freedom and determine the true DOF of the system by logically analyzing how the coordinate frames cancel out each other. We will use Figure 5.4 as our reference diagram and analyze the DOF of our system as follows:

1. Since we are allowed to choose the world coordinate frame at any location, we will fix it at an arbitrary point along the axis of rotation of the Dihedral group. A special location on this axis will become evident in the next few steps. This maneuver eliminates two DOF.

2. Now, let us consider the camera pose ${}_{w}\mathsf{T}_{c}$. It is evident that increasing the scale of the structure while moving the camera farther away, and vice versa creates the same net measurements of points in the image. We can fix the scale of the system by ensuring that the camera is at a fixed distance from the world coordinate frame. We choose this distance to be 1 unit as shown in Figure 5.4. The camera can rotate and translate while remaining at this distance from the origin of the world. This eliminates 1DOF.

3. The constraint that we have imposed on the camera essentially allows it to move on a unit sphere of radius 1. However, we can fix an additional degree of freedom by considering the relative rotation of the world coordinate frame and the position of the camera on this unit sphere. Changing both these in tandem creates the same net measurement in the image. We can therefore exploit this relative constraint by fixing one of them. For example, if we fix the location of the camera at a $(0, -1, 0)$ this essentially fixes both the rotational components of the world coordinate frame such that $z-$axis is coincidental with the axis of rotation and the $y-$axis pointing away from the camera as shown in Figure 5.4. This eliminates 4 DOF.

4. Lastly, we need to determine the extra DOF arising as a result of allowing the points in the asymmetric unit to exist in their own coordinate frame. Since we are allowed

to choose this, we can pick the location of the origin of the asymmetric unit at the same location of the world origin. In addition to this, definition of the Dihedral symmetry group fixes the orientation of the asymmetric unit by only allowing its axis to be coincidental with the axis of rotation. However, it is free to rotate about this axis and its orientation is indeed a free parameter. This eliminates 5 DOF.

After fixing the free parameters, the measurement function given in Equation 5.10 takes the following form:

$$\mathsf{z}_k = \Pi\left(\left({}_w\mathsf{R}_c\right)^{\mathsf{T}}\ \mathsf{R}_z\left(\theta\right)\left(\mathsf{g}_1^{\mathsf{w}_j(1)}\mathsf{g}_2^{\mathsf{w}_j(2)}\mathsf{v}_{\mathbf{p}_{\mathbf{j}_k}}\right)+\mathsf{t}\right) \tag{5.14}$$

where $\mathsf{R}_z\left(\theta\right)$ represents an unknown rotation about the $z-$axis for the orientation of the asymmetric unit. The relative translation of the asymmetric unit ${}_w\mathsf{t}_a$ is eliminated and $\mathsf{t} = \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}$ corresponding to the fixed location of the camera. The unknown parameters of the system are $\theta, N, {}_w\mathsf{R}_c$. The net DOF of the system is given by $5 + K + 3 * |\mathcal{V}|$ where K is the number of DOF of the intrinsics of the camera and $|\mathcal{V}|$ is the number of points in the asymmetric unit. The variable N is discrete, however, we choose a continuous parametrization for this angle such that $\beta \approx \frac{360}{N} \in \mathbb{R}$.

$$\left(\beta^\star, \theta^\star, {}_w\mathsf{R}_c^\star, \mathcal{V}^\star, K^\star\right) = -\log\left(\mathcal{L}\left(K, \mathcal{V}, \beta, \theta, {}_w\mathsf{R}_c; \mathcal{Z}, J, \mathsf{P}, \mathcal{W}\right)\right) \tag{5.15}$$

Single Image Reconstruction with Lattices

Figure 5.5: **Single Image Reconstruction with lattices**: Example of Complex 3D symmetry with lattice: The symmetry group of windows of the Leuven Stadius building belongs to the Layer group Pm. In this section we will explore the reconstruction of this building by rigorously modeling the symmetry group Pm using the generative model introduced in the previous Chapters.

In this section I provide a generative model for a camera imaging a complex 3D symmetry **with lattice** by using the generative model given in Chapter 3,. In this section, I will fo-

cus on the reconstruction of the Leuven Stadius building, the very same we considered in Chapter 4. However, to demonstrate our approach, we choose to model this building as a 1D lattice in 3D space, with the repetitions of windows occurring in the horizontal direction as the symmetry of focus. Since the windows themselves exhibit bilateral symmetry, their collective symmetry belongs to the group Pm.

In this section I will describe the generative model of a Layer group Pm. I introduced the 2D analogue of this symmetry in Chapter 3 and I extend that to the 3D case here. The generative model for this case is shown in Figure 5.6(a) and the measurement prediction function is given by

$$\mathsf{x}_j = h_{31}\left(\mathsf{v}_p, \mathsf{u}_p, \mathsf{g}_1, \mathsf{w}_j, \mathsf{B}, \mathsf{q}_j, \mathsf{T}\right) + \eta_{30} \tag{5.16}$$

where $h_{31}\left(\cdot\right)$ is given by

$$h_{31}\left(\cdot\right) = \begin{cases} {}_w\mathsf{R}_a\mathsf{B}\left(\mathsf{g}_1^{\mathsf{w}_j(1)}\begin{bmatrix}\mathsf{u}_{\mathbf{p}_j}\\ \mathsf{v}_{\mathbf{p}_j}\end{bmatrix} + \begin{bmatrix}\mathsf{q}_j\\ 0\end{bmatrix}\right) + {}_w\mathsf{t}_a & \mathsf{w}_j\left(0\right) = c \\ {}_w\mathsf{R}_a\mathsf{B}\left(\mathsf{g}_1^{\mathsf{w}_j(1)}\begin{bmatrix}0\\ \mathsf{v}_p\end{bmatrix} + \begin{bmatrix}\mathsf{q}_j\\ 0\end{bmatrix}\right) + {}_w\mathsf{t}_a & \mathsf{w}_j\left(0\right) = b \\ {}_w\mathsf{R}_a\mathsf{B}\left(\mathsf{g}_1^{\mathsf{w}_j(1)}\begin{bmatrix}0.5\\ \mathsf{v}_p\end{bmatrix} + \begin{bmatrix}\mathsf{q}_j\\ 0\end{bmatrix}\right) + {}_w\mathsf{t}_a & \mathsf{w}_j\left(0\right) = a \end{cases} \tag{5.17}$$

where

$$\mathsf{g}_1 = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5.18}$$

$\mathcal{W} = \{\mathsf{w}_p\}_{j=1}^{n} \left| \mathsf{w} \in \{\{a,b,c\},\{0,1\}\} \right.$ is the Wyckoff indicator variable, $({}_w\mathsf{R}_a, {}_w\,\mathsf{t}_a) \in \mathbb{SE}(3)$ is the pose of the asymmetric unit with respect to the world, $0 \leq \mathsf{u}_p < 0.5$ and $\mathsf{v}_p \in \mathbb{R}^2$, $\mathsf{q}_j \in \mathbb{Z}$

$$\mathsf{B} = \begin{bmatrix} b & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

which reduce to the following simple equation for points in the general position:

$$h_{31}(\cdot) =_w \mathsf{R}_a \mathsf{B} \left(\mathsf{g}_1^{\mathsf{w}_j(1)} \begin{bmatrix} \mathsf{u}_{\mathbf{p}_j} \\ \mathsf{v}_{\mathbf{p}_j} \end{bmatrix} + \begin{bmatrix} \mathsf{q}_j \\ 0 \end{bmatrix} \right) +_w \mathsf{t}_a \tag{5.19}$$

Similar to the above case,we model in the projection function of the camera to obtain a generative model for camera measurements shown in Figure 5.6(b) which corresponds to

the following measurement prediction equation

$$h_{31}(\cdot) = \Pi \begin{cases} ({}_{w}\mathsf{R}_{c})_{w}^{\mathsf{T}}\, \mathsf{R}_{a}\mathsf{B} \left(\mathsf{g}_{1}^{\mathsf{w}_{j}(1)} \begin{bmatrix} \mathsf{u}_{\mathbf{p}_{j}} \\ \mathsf{v}_{\mathbf{p}_{j}} \end{bmatrix} + \begin{bmatrix} \mathsf{q}_{j} \\ 0 \end{bmatrix} \right) + ({}_{w}\mathsf{R}_{c})_{w}^{\mathsf{T}}\, \mathsf{t}_{a} - {}_{w}\,\mathsf{t}_{c} & \mathsf{w}_{j}(0) = c \\ ({}_{w}\mathsf{R}_{c})_{w}^{\mathsf{T}}\, \mathsf{R}_{a}\mathsf{B} \left(\mathsf{g}_{1}^{\mathsf{w}_{j}(1)} \begin{bmatrix} 0 \\ \mathsf{v}_{p} \end{bmatrix} + \begin{bmatrix} \mathsf{q}_{j} \\ 0 \end{bmatrix} \right) + ({}_{w}\mathsf{R}_{c})_{w}^{\mathsf{T}}\, \mathsf{t}_{a} - {}_{w}\,\mathsf{t}_{c} & \mathsf{w}_{j}(0) = b \\ ({}_{w}\mathsf{R}_{c})_{w}^{\mathsf{T}}\, \mathsf{R}_{a}\mathsf{B} \left(\mathsf{g}_{1}^{\mathsf{w}_{j}(1)} \begin{bmatrix} 0.5 \\ \mathsf{v}_{p} \end{bmatrix} + \begin{bmatrix} \mathsf{q}_{j} \\ 0 \end{bmatrix} \right) + ({}_{w}\mathsf{R}_{c})_{w}^{\mathsf{T}}\, \mathsf{t}_{a} - {}_{w}\,\mathsf{t}_{c} & \mathsf{w}_{j}(0) = a \end{cases} \tag{5.20}$$

where $\Pi, ({}_{w}\mathsf{R}_{c}, {}_{w}\,\mathsf{t}_{c}) \in \mathbb{SE}(3)$ have the same meaning as in the previous case. The form of the equation for the general position is

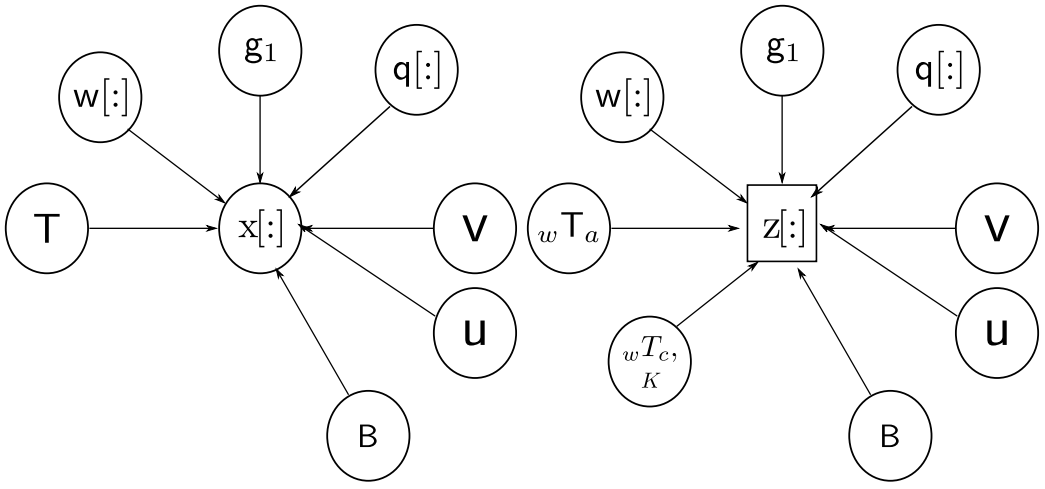
$$h_{31}(\cdot) = ({}_{w}\mathsf{R}_{c})_{w}^{\mathsf{T}}\, \mathsf{R}_{a}\mathsf{B} \left(\mathsf{g}_{1}^{\mathsf{w}_{j}(1)} \begin{bmatrix} \mathsf{u}_{\mathbf{p}_{j}} \\ \mathsf{v}_{\mathbf{p}_{j}} \end{bmatrix} + \begin{bmatrix} \mathsf{q}_{j} \\ 0 \end{bmatrix} \right) + ({}_{w}\mathsf{R}_{c})_{w}^{\mathsf{T}}\, \mathsf{t}_{a} - {}_{w}\,\mathsf{t}_{c} \tag{5.21}$$

The optimal value of the unknown parameters of the system is given by

$$(c^{\star}, \mathcal{V}^{\star}, \mathcal{U}^{\star}, \Theta^{\star}, \mathsf{B}^{\star}, {}_{w}\,\mathsf{T}_{a}) = \max_{(c, \mathcal{V}, \mathcal{U}, \mathsf{B}, \Theta)} \mathcal{L}(c, \mathcal{V}, \Theta; \mathcal{Z}, J, \mathsf{P}, \mathcal{W}) \tag{5.22}$$

Gauge Freedom

Similar to the above case, we can reduce the redundant degrees of freedom of the system by analyzing the geometry of the scene. Unlike the case of the Dihedral group, the group



(a) Generative Model for 3D point x_j (b) Generative Model for measurement z_k

Figure 5.6: **Generative Model for the Layer group** Pm:

Pm does not have a globally unique axis that we can leverage to analyze the system. Figure 5.6 shows the example of the Gauge freedom assosiated with the Pm symmetry group. Similar to previous case, we will analyze the geometry of the system to determine the redundant DOF a follows.

1. Under the assumption, that the camera is imaging the scene fronto-parallely, we choose the dominant 2D symmetry plane of the system as the one that includes the axis of the lattice along with the component of $\mathsf{v} \in \mathbb{R}^2$ that has the highest variance, in this case, this is the facade of the building. We fix the $x-$axis of the world to coincide with axis of the lattice and the $y-$axis to coincide with the dominant axis of v. The location of the frame is allowed to vary on this plane, a unique location will be evident with subsequent analysis.

2. Now, let us consider the camera pose ${}_w\mathsf{T}_c$. Similar to the previous scenario, we can

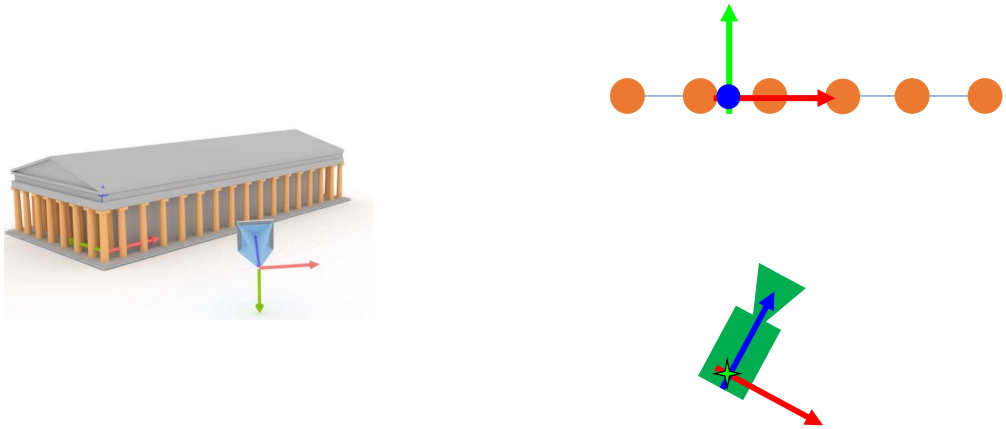


Figure 5.7: **Gauge freedom for camera imaging a scene exhibiting** Pm : The 1-dimensional lattice group contains 3 Wyckoff positions. The choice of coordinate frame for the camera and the world shown in this figure minimizes the number of redundant DOF of the system. In Section 5.3.3, I discuss in detail the degrees of freedom corresponding to this geometric configuration. The total gauge freedom for this configuration is given by $(5 + |K|)$

fix the distance to be at 1 unit from the 2D plane determined above. The camera can rotate and translate while remaining at this distance from the origin of the world. This eliminates 1DOF.

3. Since the location of the camera and the location of the world coordinate frame can translate parallely, there is a redundant degree of freedom and fixing one of these essentially fixes the other. We chose the location of the world origin such that it is directly in front of the camera and located on the facade. Therefore, by fixing the location of the camera at $(0, -1, 0)$, a unique coordinate system for the camera and the world emerges as shown in Figure 5.7.

4. Since the pose of the asymmetric unit ${}_w\mathsf{T}_a$ relative to the world coordinate frame is a free parameter, we can eliminate the free parameters assosiated with this variable

by fixing the rotational component, ${}_w\mathsf{R}_a$ at I_3, making it coincidental with the axes of the world. We can fix one free parameter of the translation by choosing it to be located on the building facade. We can fix one more parameter of this translation by choosing the y-component to be at the same height as the camera and the world origin. which leaves the translation of the asymmetric unit parallel to the lattice axis as a free parameter.

After fixing the free parameters, the measurement function given in Equation 5.20 takes the following form:

$$h_{31}(\cdot) = \Pi \begin{cases} ({}_w\mathsf{R}_c)^{\mathsf{T}}\,\mathsf{B}\left(\mathsf{g}_1^{\mathsf{w}_j(1)}\begin{bmatrix}\mathsf{u}_{\mathbf{p}_j}\\ \mathsf{v}_{\mathbf{p}_j}\end{bmatrix}+\begin{bmatrix}\mathsf{q}_j\\ 0\end{bmatrix}\right)+({}_w\mathsf{R}_c)_w^{\mathsf{T}}\,\mathsf{t}_a-\mathsf{t} & \mathsf{w}_j(0)=c \\ ({}_w\mathsf{R}_c)^{\mathsf{T}}\,\mathsf{B}\left(\mathsf{g}_1^{\mathsf{w}_j(1)}\begin{bmatrix}0\\ \mathsf{v}_p\end{bmatrix}+\begin{bmatrix}\mathsf{q}_j\\ 0\end{bmatrix}\right)+({}_w\mathsf{R}_c)_w^{\mathsf{T}}\,\mathsf{t}_a-\mathsf{t} & \mathsf{w}_j(0)=b \\ ({}_w\mathsf{R}_c)^{\mathsf{T}}\,\mathsf{B}\left(\mathsf{g}_1^{\mathsf{w}_j(1)}\begin{bmatrix}0.5\\ \mathsf{v}_p\end{bmatrix}+\begin{bmatrix}\mathsf{q}_j\\ 0\end{bmatrix}\right)+({}_w\mathsf{R}_c)_w^{\mathsf{T}}\,\mathsf{t}_a-\mathsf{t} & \mathsf{w}_j(0)=a \end{cases} \tag{5.23}$$

where $\mathsf{t}=\begin{bmatrix}0\\ -1\\ 0\end{bmatrix}$, ${}_w\mathsf{t}_a=\begin{bmatrix}t_x\\ 0\\ 0\end{bmatrix}$ while the other parameters have no additional constraint. The net DOF of the system is given by $5+K+2*|\mathcal{V}|+|\mathcal{U}|$ where K is the number of DOF of the intrinsics of the camera and $|\mathcal{V}|, |\mathcal{U}|$ is the number of points in the asymmetric unit. The optimization problem essentially reduces to minimizing the negative log-likelihood

with the following free parameters:

$$(\mathsf{B}^{\star},_{w}\,\mathsf{t}^{\star}_{a},_{w}\,\mathsf{R}^{\star}_{c},\mathcal{V}^{\star},\mathcal{U}^{\star},K^{\star}) = -\log\left(\mathcal{L}\left(K,\mathcal{V},\mathcal{U},_{w}\,\mathsf{t}_{a},_{w}\,\mathsf{R}_{c};\mathcal{Z},J,\mathsf{P},\mathcal{W}\right)\right) \tag{5.24}$$

Determining Symmetric Features in an Image

In this section, we address the problem of determining symmetric features in an image. This is equivalent to determining the measurements $\mathsf{z}_{\mathbf{j}_k}$ and assigning Miller indices q_j and the Wyckoff Variable w_j and the index of the asymmetric unit $\mathbf{p}_{j_k}$ for a sufficient set of symmetric 2D features in the image. This was modeled using the correspondence information J and P.

Problem Statement

Given an image $\mathcal{I}$ along with a set of 2D features $\tilde{\mathcal{Z}}$ and its descriptors $\tilde{\mathcal{D}}$ and the symmetry scene exhibits; our goal is to generate a set of measurements $\mathcal{Z}$ along with the complete correspondence information from measurements to 3D points to asymmetric units modeled by (J,P), Wyckoff positions $\mathcal{W}$ and in the case of lattice, the Miller indices $\mathcal{Q}$. This is an incredibly hard problem to solve without any notion of semantic information. We provide one solution to this problem below that obtains some of the semantics via a manual intervention stage.

We solve the problem of determining measurements and correspondence information using a **single manual intervention** stage where a user is presented with the image and is

requested to draw a single **rectangular** region on top of a repeating pattern in the image. Since we deal with architectural data in this thesis, windows are one of the most prominent repetitions that occur in buildings. However, the interpretation of what the window signifies, depends on the symmetry that we are trying to exploit to determine the 3D structure. I will therefore analyze the ideal manual intervention for the two symmetry groups that we have described above.

Figure 5.8: **Ideal Manual Intervention**: Here we demonstrate Ideal manual intervention on two datasets that has been under consideration for this chapter. [**Top Row**] The Royal Albert Hall dataset, where the points on the top row of the window belong to the Dihedral symmetry group. Actually, the bottom row window and balcony on top also belong to the dihedral group (We will see how to determine this in the next section). The Ideal manual intervention is shown here as the user draws a rectangle with blue borders colored with green. This is ideal because this window is least ambiguous for determining repeated patterns across the image in the next step. Since the window exhibits reflective symmetry, we know that this corresponds to twice the angular displacement of the the asymmetric unit [**Bottom Row**] The Leuven dataset is the dataset with symmetry and lattice. Here the window is one unit cell of the lattice. Similar to he Royal Albert hall, the window center corresponds to the plane of reflection.

Ideal Manual Intervention Based on Symmetry type A user is generally unfamiliar with the details of symmetries of a scene. The concept of symmetry is often vague and most people understand it as some form of repetition. When it comes to architectural symmetry, it is clear that it exhibits many repetitions around the windows and there is "some form of symmetry" there. We define an ideal manual intervention as that which encompasses all possible known Wyckoff positions. In the case of lattice, an ideal manual intervention would be to segment out the unit cell. The user would also be advised to pick a region that is least distorted by the projective mapping and radial distortion of the camera. This is because, in the next step we rely on appearance-based feature matching and these kinds of techniques work best when there are minimal appearance changes between the regions that are being matched. Figure 5.8 shows an example of ideal feature matching for the two datasets that we are considering, the Royal Albert Hall (Dihedral symmetry group D_N with unknown rotation and having a fronto-parallel image of the building) and the Leuven Town Hall building (lattice symmetry Pm that has both translational symmetry and reflection symmetry).

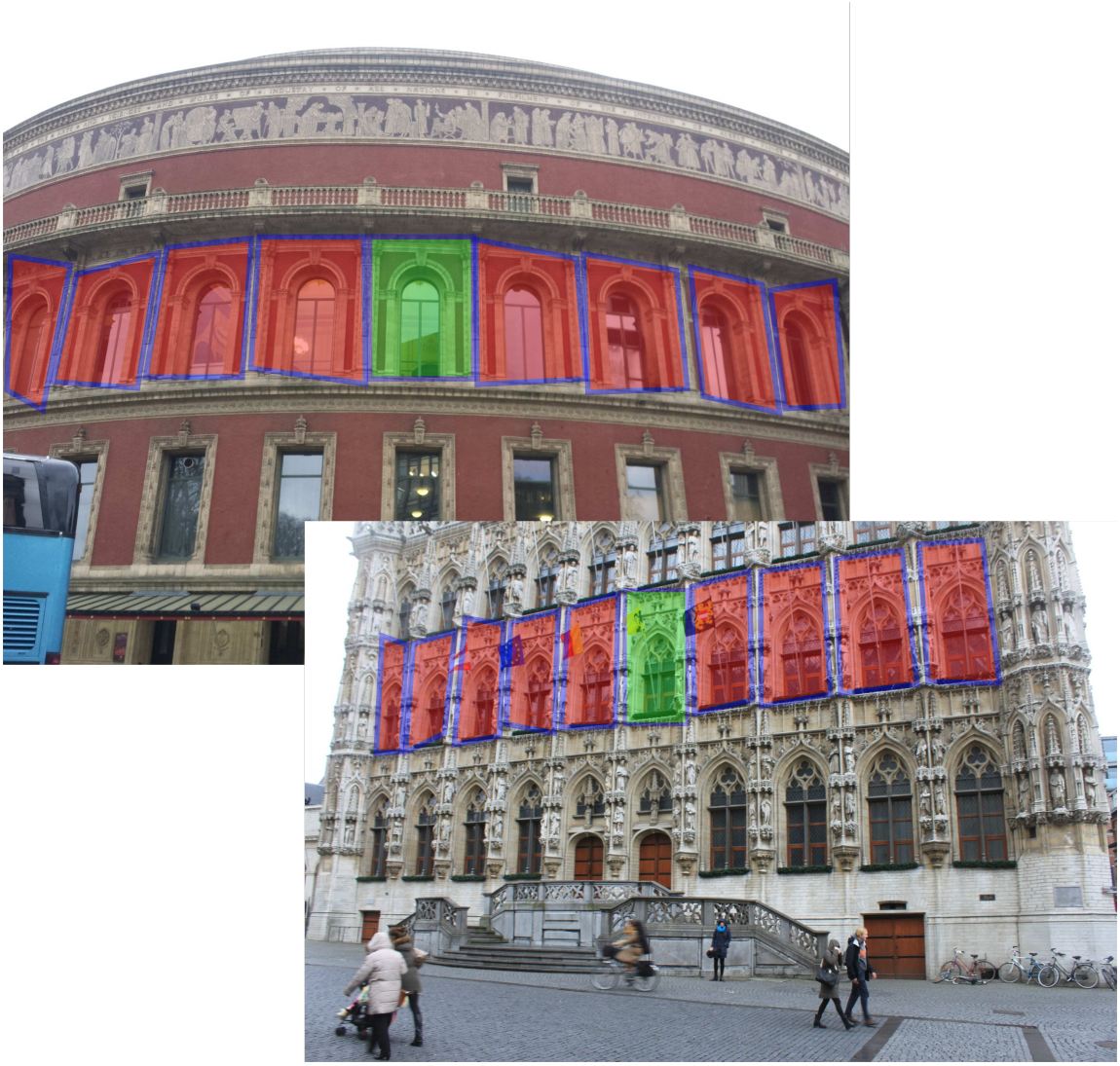


Figure 5.9: **Automatic Identification of Repeated Regions using Homography**: Given that a user has input a bounding box around the region of interest (shown in green), we compute matching regions using an automatic process. The details of the algorithm are given Section 5.3.2.2. This is essentially a Homography-driven algorithm which assumes that the user drawn region is roughly planar and find regions in the image that map this bounding box by a 3×3 Homography matrix.

In this section we will detail an algorithm which takes as an input, a user provided manual region described in the previous section and output all the locations of that region repeated within the image. Such kind of algorithms were popular in early computer vision literature such as Leung *et al.* [LM96]. More recently, there have been more successful versions of these algorithms such as [LLH04]. Here we will use a much more primitive algorithm that is more restrictive than competing ones that achieve the same goal. This is because our goal is not to implement the best region recognition algorithm, but instead, the one that can get the initial process of optimization started, after which we can use guided matching to determine more matches accurately.

1. The **first step** of our algorithm after the user drawn rectangle, is to find approximately similar regions in the image that correspond in appearance to the user input. For this we use a normalized cross correlation method with a Sum of Absolute Difference metric to determine matching regions within the image. If the user drawn rectangle spans from $(m - w/2, n - w/2)$ to $(m + w/2, n + h/2)$, we compute the SAD value for every pixel in the image (after smoothing the image) with respect to the region drawn by the user as:

$$\forall\,(c_i, c_j) \in \mathcal{I}, \sum_{i=-w/2, j=-h/2}^{i=w/2, j=h/2} \left|\mathcal{I}\,(c_i + i, c_j + j) - \mathcal{I}\,(m + i, n + j)\right|$$

   where c_i, c_j represents a pixel location in the image. The rectangle that is drawn has a width w and height h and its center is at location (m, n).

2. In the **second step,** we collect the peaks of the SAD function. We find the top **two peaks** that are approximately near at $(m - w, n)$ and $(m + w, n)$. With the

exception of few cases, these peaks typically correspond to the maximum value of SAD image.

3. In the **third step**, we translate the user input rectangle such that their centers lie at the two peaks that are identified in step two. We do not alter the size of the rectangles in any way.

4. In the **fourth step,** we compute putative feature matches between features that fall within the original rectangle and the translated rectangle. We use approximate nearest neighbors with $N = 5$.

5. In the **fifth step**, we compute two Homography matrices H_l, H_r that correspond to transformation of the input rectangle to the right and to the left of the user drawn rectangle. Given putative matches (after preprocessing the points as detailed in [HZ03]), we briefly detail the process of robust estimation of Homography between a pair of regions in a RANSAC loop.

   (a) Select 4 putative matches, compute an initial homography between the regions

   (b) Compute inliers such that the transformed point is within an ϵ radius of its corresponding point

   (c) If current inlier set is greatest, keep it, if not discard and try again.

   (d) Estimate Homography using DLT using the set of inliers.

6. In the **sixth step**, we repeat all the above steps to the right and to the left of the newly estimated quadrilaterals. (for SAD case, we use the smallest rectangle that can fit inside the quadrilateral as the template.)

Mapping Repeated regions to Optimization Variables

In this section we discuss an algorithm that takes a set of repeated regions in an image and determines the variables that we assume to be known in our objective functions (Equations 5.24&5.15). In particular, for all measurements in $\mathcal{Z}$, (1) correspondence between measurement z_k and 3D point x_j, J (2) Correspondence between a point in the asymmetric unit u_p and 3D point x_j, P (3) all observed Wyckoff position if possible $\mathcal{W}$ (4) All Miller indices $\mathcal{Q}$ if the scene exhibits lattice symmetry.

Dihedral Symmetry Group For the Royal Albert Hall exhibiting dihedral symmetry D_N, the user marked region around the window represents twice the asymmetric unit that has been obtained by the known generator i.e the reflection generator g_1. The single window therefore represents the same Wyckoff position for the second generator in the system g_2 which corresponds to the rotational symmetry of the Dihedral group. Because this is not known, we can only make relative prediction about the Wyckoff position by assuming that one of the window represents the 0^{th} position. Windows adjacent to each other are separated by a single unit in the Wyckoff assignments. Window to the left of the current window are treated at -1 and windows on the right are treated at $+1$ to the assigned Wyckoff position of the current window.

In order to determine the Wyckoff positions that correspond to reflection symmetry, we divide the input region represented by the homography -determined quadrilateral into vertically equivalent region. If the quadrilateral is specified by the point set

$$Q_1 = (\{x_1, y_1\} \ldots \{x_4, y_4\})$$

we split this into two equivalent quadrilaterals

$$Q_{N1} = (\{x_1, y_1\}, \{x_4, y_4\}, \{(x_1 + x_2)/2, (y_1 + y_2)/2\}, \{(x_3 + x_4)/2, (y_3 + y_4)/2\})$$

and

$$Q_{N2} = (\{x_3, y_3\}, \{x_4, y_4\}, \{(x_3 + x_4)/2\,(y_3 + y_4)/2\}, \{(x_3 + x_4)/2, (y_3 + y_4)/2\})$$

We use all features that are deemed inliers in the RANSAC-Homography stage and obtain matches between the two feature that are at an equal distance from the bordering line between the two quadrilaterals. These are mirror symmetric features and are assigned a Wyckoff position of 0 or 1 depending on our choice. We have to ensure that given that we make a choice and be consistent with this; for example if we choose all points on the left quadrilateral as 0, then we choose all the points on the right in all the regions as 1 (meaning they are reflected points). This will assign the Wyckoff parameters corresponding to reflection symmetry for all measurements z_k. We have so far determined the Wyckoff variable w_j for a particular measurement z_k in the image. We assume that every point that is being observed has a corresponding unique point in 3D. Therefore the correspondence information between measurement $\mathsf{z}_k$and 3D point x_j is known.

In order to determine the correspondence variable P, we need to establish feature "tracks", which are matches between features across multiple quadrilateral regions obtained from linking pairwise matches between regions. To draw an analogy, in SfM, we use a RANSAC and geometric filtering stage at this junction to convert feature matches into tracks of feature matches across multiple images. In the case of symmetry, this is not possible because

of the model complexity of the generative model of symmetry is high (we described the number of free parameters in the previous sections). RANSAC tends to perform best when the underlying model parameters are low expressed best by a parameter called co-dimension. The co-dimension states that, with increasing model complexity, the percentage of inliers has to increase for a successful completion of the RANSAC algorithm. For the types of symmetry models under consideration, we need a inlier percentage in excess of 99% for RANSAC to converge. Practically, this is not possible we therefore rely on appearance based methods to determine feature tracks. Every adjacent quadrilateral already has 4 features that match either directly or via reflective symmetry to one another; 2 of which are within the same quadrilateral and the other two with the adjacent quadrilateral, thanks to matches established during the RANSAC loop of Homography. We start with this 4 feature match and look in the next adjacent quadrilateral. If it so happens that the same feature that is matched in the current quadrilateral pairs also has a match in the adjacent quadrilateral pairs, we chain the two pairs together to make a set of 6 features. If however the current quadrilateral pairs do not have a corresponding feature match in the adjacent quadrilateral pairs, one of the following possible scenarios could be true (1) The current feature pairs is the entire measurement set for this track because of occlusions and camera FOV limitations or, (2) The current feature pairs do not have a valid match that survived the RANSAC loop of Homography. We cannot do anything for the first case, except create a set of virtual measurements to see if there is match in the next stage. We will get describe this in the next section and focus how we can address the second case here. For the second case, if there are two feature pairs that are spatially close to each other (within a radius r), we see if we can link these two together by checking the corresponding de-

scriptor values of the edge features. We compare both the original and the reflected feature to ensure no incorrect matches. If both the original and reflected features are a match, we link the feature pairs together into a set of 6 measurements. The point of linking is taken as the average location of the features that are being matched. In case there is no match, we create a virtual measurement with the descriptor value of the previous feature to see if there is a match in the next region. This would result in the case of missing measurements. Every feature track that we identify (including the ones with missing measurements) is assigned a unique id, p and this is the index of the point in the asymmetric unit. In every track we also know the measurement index and the point index j . Therefore, we have established the variable P.

Group Pm Additionally when we have lattices, we need to establish the Miller Indices $\mathcal{Q}$. In the case of the group Pm which is similar to the Dihedral group that we discussed earlier in terms of having the same reflection generator, we proceed exactly as above. Except that the second Wyckoff indicator variable here is replaced by the miller index q, since this pertains to a translational lattice rather than a rotation like in the case of the group D_N. I have demonstrated that we can determine all the optimization parameters $(J, \mathsf{P}, \mathcal{Q}, \mathcal{W})$ for both the above cases using appearance based feature matching alone.

Guided Matching

In this section, I will provide a guided matching that increases the density of the points in the asymmetric unit by increasing the number of measurements of the system. In Equa-

tions 5.7&5.17 we described the generative model that takes a point $\left(\mathsf{u}_{\mathbf{p}_j}, \mathsf{v}_{\mathbf{p}_j}\right)$ in an asymmetric unit to a corresponding 3D point x_j for two specific cases of symmetry groups; D_N and Pm. We can generalize the relation of a symmetric point x_j and the symmetry parameters using a general non-linear function $f(\cdot)$

$$\mathsf{x}_j = f\left(\mathsf{u}_{\mathbf{p}_j}, \mathsf{v}_{\mathbf{p}_j}, \mathsf{w}_j, \mathsf{q}_j, \Theta\right)$$

where the space spanned by $\mathsf{u}_{\mathbf{p}_j}, \mathsf{v}_{\mathbf{p}_j}$ depends on the type of symmetry and lattice and Θ models the symmetry and lattice parameters of the system q_j and w_j models the Miller indices (if it exists) and Wyckoff positions.

A guided matching scheme exploits the optimized values of the symmetry parameters $\Theta^\star$ to determine new measurements z_k. While this sounds trivial, it is crucial to note that, a point x_j that maps to a measurement z_k is influenced by **non-global** parameters, $\mathsf{u}_{\mathbf{p}_j}, \mathsf{v}_{\mathbf{p}_j}$ which we cannot determine by appearance-based matching alone. We implement a constrained search algorithm that finds the best values of $\mathsf{u}_{\mathbf{p}_j}, \mathsf{v}_{\mathbf{p}_j}$ that generates the corresponding measurements z_k. Figure 5.14 shows the output of the guided matching algorithm. We see that the number of points has significantly increased as a result of this step.

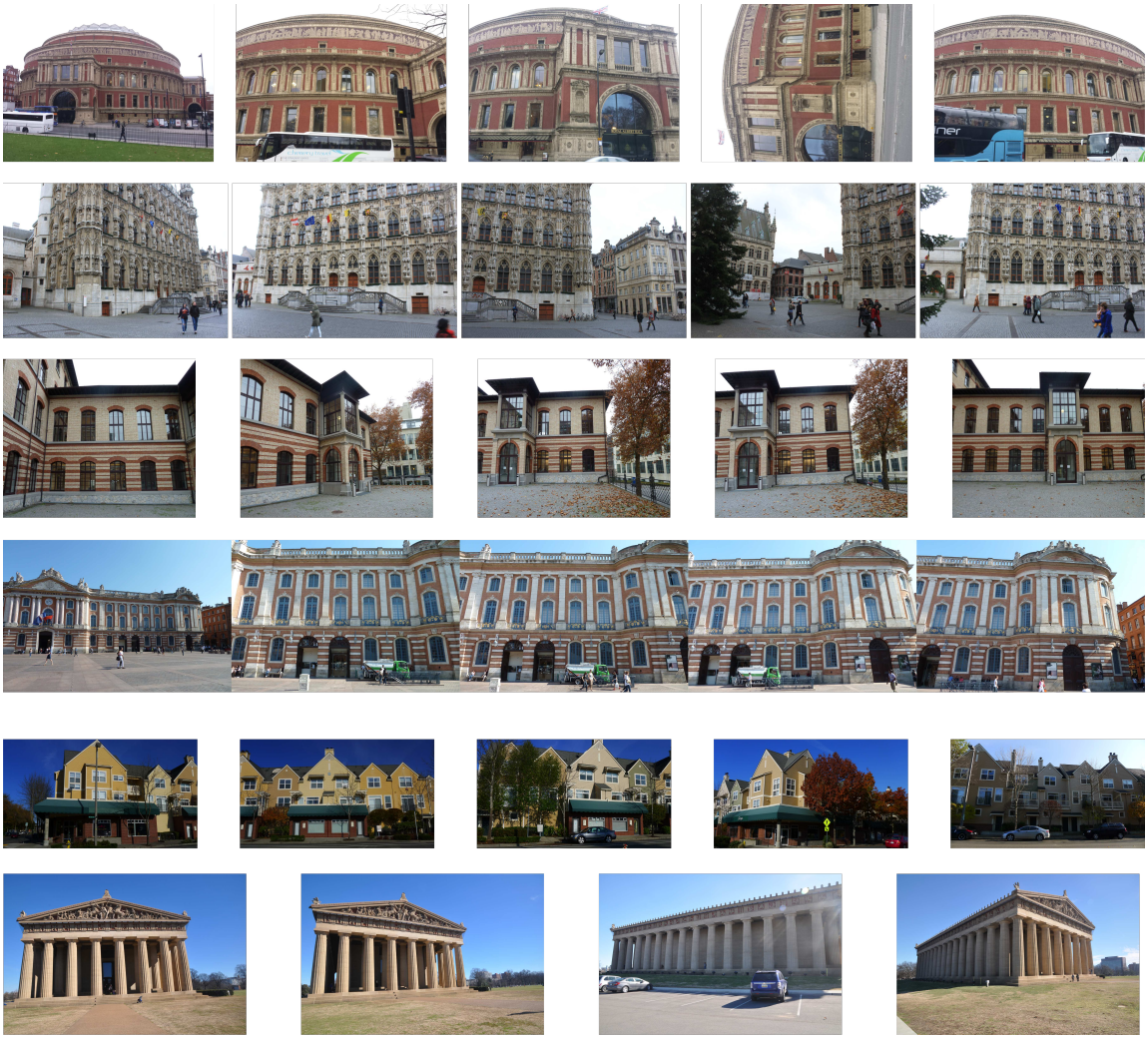


Figure 5.10: **Overview of datasets for Experiments**: I use one rotational symmetry dataset (top row) and 5 translational symmetry datasets 3 of which are from Cohen *et al.* [Coh+12] and two of which is my own. From top to bottom: The Royal Albert Hall: This is elliptical building in London but at close camera ranges it is approximately circular. The Leuven Stadhuis: A dataset to show translational symmetry. It exhibits both 1D lattice and a 2D lattice structure. Row 3,4,5 the Symmetry dataset as described in [Coh+12]. The last row.

Qualitative and Quantitative Results

Experimental Setup

In this section I validate all the theory that we have so far discussed in this chapter with experimental results. I use 4 translational symmetry datasets and two rotationally symmetric datasets. The datasets that we use is shown in Figure 5.10.

Every dataset in the above has its own unique set of challenges. The Royal Albert Hall is an ellipse in plan, with major and minor axes of 83 m (272 ft) and 72 m (236 ft); an eccentricity of 0.86, although we treat it as approximately circular in our analysis of it. This is because the majority of the pictures on which we operate is less than 15m from the building at this distance the approximation of circular nature of the building is not too incorrect. We still choose to utilize this dataset because it has a particularly rich feature set which lends well into matching algorithms that we use. In the second row, I show the LEUVEN dataset. The Leuven dataset is an image of the Leuven Stadhuis building. It is extremely feature rich with a lot of symmetry. The main features of interest to us are the windows and the ornate architecture surrounding it; There are 10 repetitions in the windows in the horizontal direction. The top two rows of windows can be seen a 2D lattice but if all the 3 rows of windows are considered together, it is a 1D lattice. This is a challenging dataset and it is a good test to gauge if our structure estimation is able to finely distinguish the subtle gradations in depth values. Both the Leuven and the Royal Albert hall datasets, while exhibiting a high degree of symmetry also have a lot of non-symmetric regions that allow us to localize cameras approximately; The Royal Albert hall

Figure 5.11: **Preprocessing of data**: Extracting of features and removal of radial distortion in the images. Given an image we first extract SIFT features and descriptors. For the Royal Albert Hall and the Leuven datasets, the number of SIFT features that are detected range in the order of about 10000. For the relatively feature deprived environments, from [Coh+12], we get around 8000 SIFT features. The images are undistorted and the the location of features suitable transformed.

has unique Frieze patterns at the top of its circular structure which is particularly useful for localizing cameras. This is required in part two when we show results on multi view structure estimation using symmetry.

Input and Preprocessing of data

The goal here is structure estimation $\mathcal{X}$ from a single view. In the previous section we described the optimization procedure and the input the algorithm requires. The measurements are image features. The extracted SIFT features for some of the above image is shown in Figure 5.11. In addition to this the input images often have severe radial distortion. We can address this issue in one of two ways. We can either model the radial distortion effect into our projection model or we can choose to rectify the image before hand. In the final result, we show results of both these cases. For now, we proceed by first pre-rectifying the image. We achieve this by using Bundler [Sna+10]. We take about 10 images and perform a local structure from motion. Since all my images are imaged by a camera having the same intrinsic parameters, we get an optimized camera intrinsic matrix

that is used for the pre-rectification. We model the radial distortion using a 3^{rd} order polynomial and the resulting image is shown in Figure 5.11. All the input features are similarly transformed by this value and we retain the descriptors computed in the original image.

Initial Structure Estimation

Figure 5.12: **Structure and Symmetry Estimation from a single image**[Leuven Dataset]: A qualitative evaluation of our approach on both rotational and translational data. The points shown in yellow are the unreflected points of the asymmetric unit and the points shown in blue are the reflected points. Together they make one unit cell of the lattice. The lattice itself is a 1D lattice whose parameters are estimated and the points are suitably translated according to this estimate.

Figure 5.13: **Structure and Symmetry Estimation from a single image**[Capitol Dataset]: We show results here on the state Capitor building, a datset provided by Cohen *et al.*, who demonstrated that SfM on this dataset can suffer as a consequence of the symmetry that this scene exhibits. Here, we show that we can obtain the structure from a single image. The meaning of the colors of the points in this image as same as that described in Figure 5.12.

Given the user-aided feature matching provides us with a set of image measurements $\mathcal{Z}$, the Miller indices $\mathcal{Q}$ and the Wyckoff positions $\mathcal{W}$, we optimize for the structure of the

scene determined by the points in the asymmetric unit $\mathcal{U}, \mathcal{V}$, the camera calibration K and its rotation ${}_w\mathsf{R}_c$, the parameters of symmetry and lattice, Θ. Unlike traditional SfM motion technique which has a geometry-aided filtering of correspondences, we do not have an equivalent stage here and are therefore forced to assume that the correspondence information that we have obtained is accurate. We allow some room for error here using the Huber Norm rather than the Frobenius norm for modeling the noise of the system Equation 1.2 which incorporates some measure of robustness into the system. Figure 5.12 presents a qualitative evaluation of our approach on both rotational and translational data. The points shown in yellow are the unreflected points of the asymmetric unit and the points shown in blue are the reflected points. Together they make one unit cell of the lattice. The lattice itself is a 1D lattice whose parameters are estimated and the points are suitably translated according to this estimate.

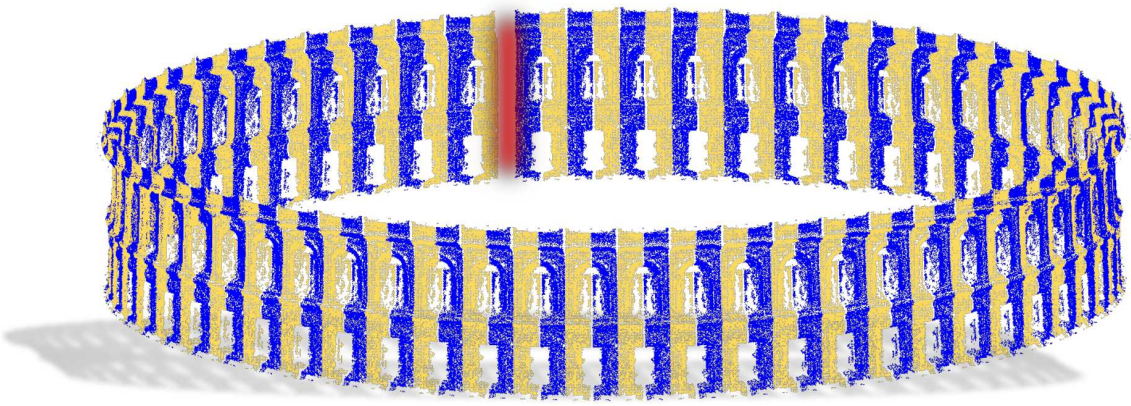
Conclusion

I conclude this chapter by summarizing the claim that this chapter has addressed. I have demonstrated that we can infer the full generative model of 3D symmetry provided in Chapter 3 by using a single image as a measurement along with a single trivial manual intervention. I have demonstrated that I can in fact extract the asymmetric unit in an object-centric coordinate system for 2 cases of 3D symmetries (1) no lattice and (2) 1D lattice. I have further shown that we can use the extracted camera pose and calibration to create virtual views and obtain a dense PMVS style reconstruction that rival SfM techniques. In the next chapter we will see how we can utilize multiple images to obtain a super-dense reconstruction that is close to a CAD like 3D model of the scene.

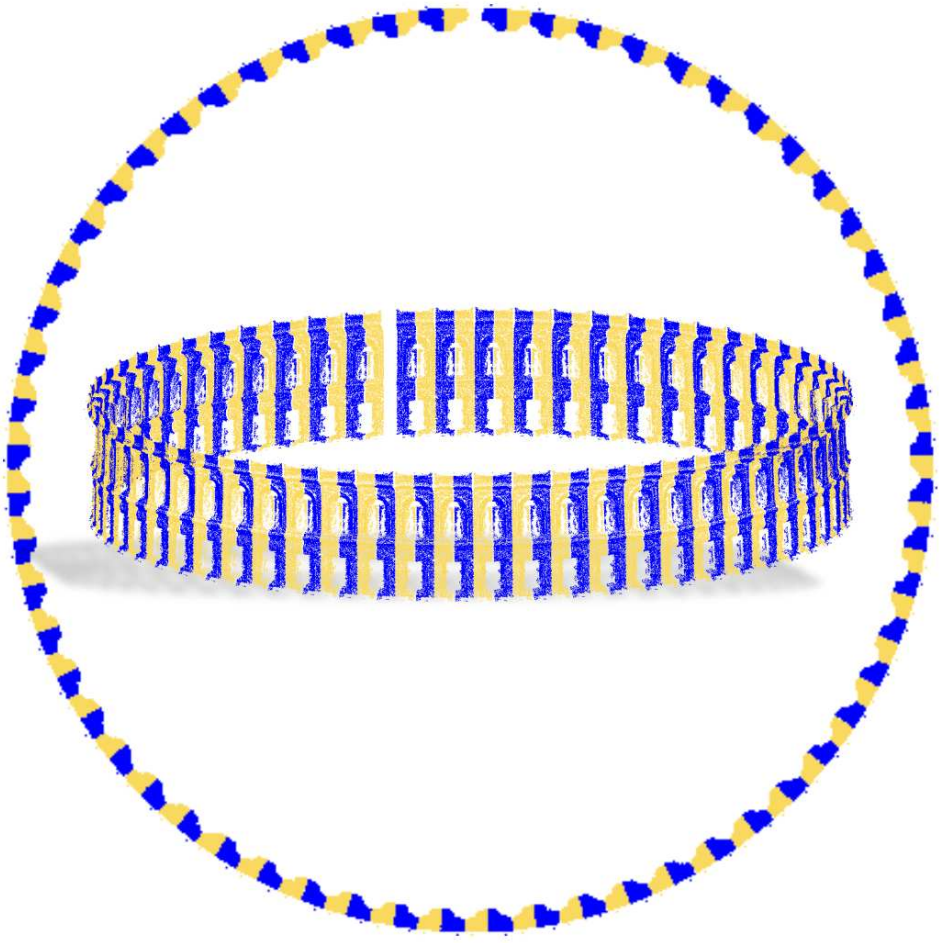
Figure 5.14: **Densification using Guided Matching**: In this figure we show the result of our densification scheme using a guided matching approach. The parameters that determine the global geometry of the system is invariant to the depth of the points, the individual measurement in the image a tightly coupled to the depth of each point. We implement a search-based guided matching scheme that determined the best value of the point u_p, v_p that generates the corresponding measurement z_k. As a result, we increase the number of measurements while also obtaining an initial estimate for these points which is subsequently optimized in the next stage.

Figure 5.15: **Output of the full pipeline:** In this figure we show the result of our full pipeline from a single user input to a densified 3D structure. I will briefly enumerate the steps taken to achieve these results: (1) a user marks a window region shown in Figure 5.8 (2) We estimate the correspondence information using an appearance based scheme Figure 5.9 and described in Section 5.3. (3) We increase the density of the points using a guided matching shown in Figure 5.14 and finally, (4) We re-optimize with the increased set of points after initializing the variables using the output of the guided matching scheme.

Figure 5.16: **Output of the full pipeline on the Dihedral group of Royal Albert Hall:** In this figure, we show the results on the Royal Albert hall dataset. The building is only oval, but at a close enough camera position, the curvature of the building appears approximately circular. We especially use this image, because it is feature rich and it allows us to obtain a dense set of 3D points similar to the Leuven dataset.

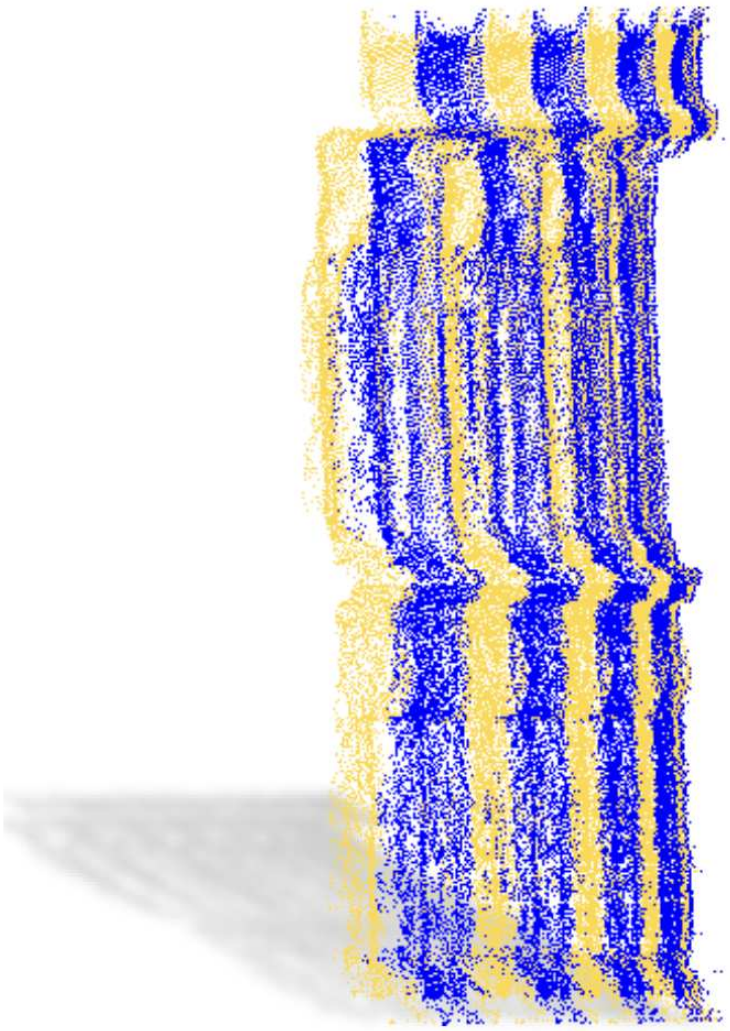


(a) Lateral view

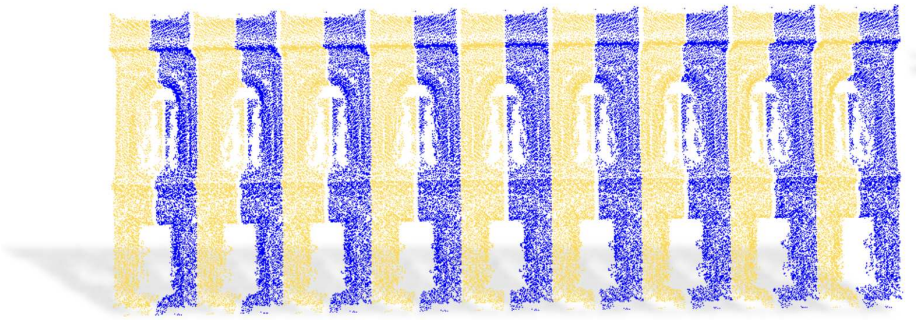


(b) Top view and lateral view

Figure 5.17: [**Single View Reconstruction: Results**]: Point cloud of the set of points extended to 61 repetitions.

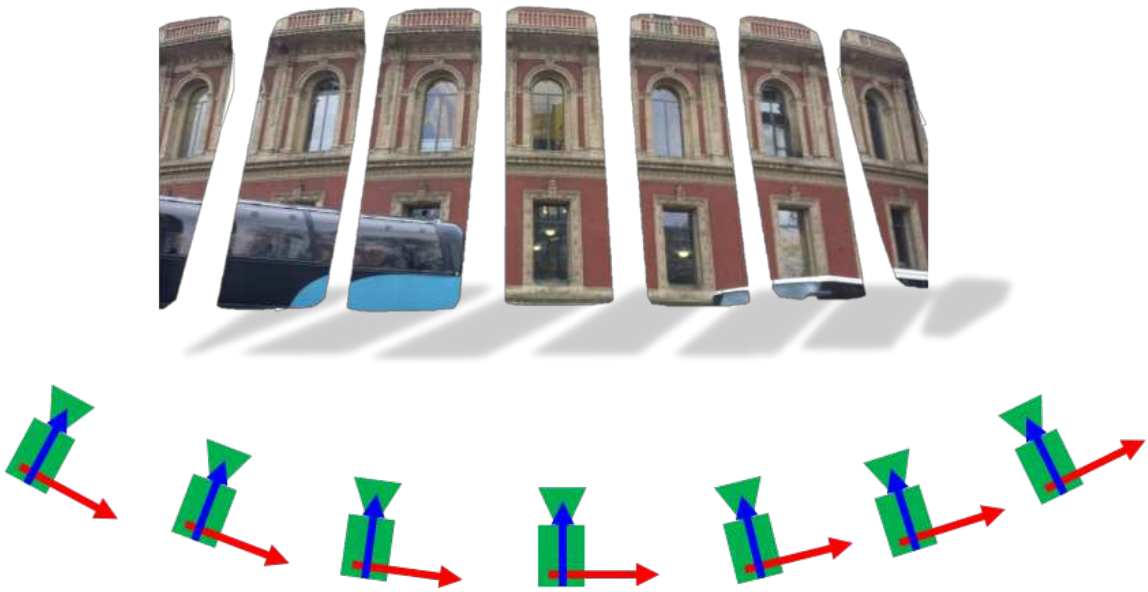


(a) Lateral view



(b) Fronto-parallel view

Figure 5.18: [**Single View Reconstruction: Results**]: Point cloud of points visible in the structure only



(a) Generation of Virtual images with virtual camera poses

(b) Densification using virtual cameras and views generated above

Figure 5.19: **Multi View stereo using a single image:** Symmetry in an image affords us a unique opportunity to further densify the structure by using a Multi-view stereo technique. This is realized by noting that transforming a symmetric structure by a value that leaves it invariant is equivalent to transforming the camera by the inverse transformation. Both these operations create the same net effect on the observed image. (a) We generate a set of virtual views using the rotational symmetry parameter that we determine for the Royal Albert Hall (b) We then use [FP10] to obtain dense reconstruction of the windows using the virtual views and the assosiated camera poses.

Chapter 6

MULTI-VIEW RECONSTRUCTION UNDER SYMMETRY

Introduction and Related work

The contents of this chapter substantiate the fifth claim of my thesis:

> *For the case of multiple views, I jointly perform structure from motion and determine the generative model of symmetry, thereby obtaining a dense 3D model of asymmetric units.*

In the previous chapter, I demonstrated how we can establish the geometry of a scene from a single image in the presence of a symmetric scene structure. Given that we know the symmetry type of the scene, we can use this knowledge to obtain the geometry of the scene up to a scale from a single user interaction with the image. In contrast to state of the art techniques that exploit simple symmetries to achieve this, I showed that by rigorously modeling complex symmetries, one can extract the full geometry of the scene along with elemental blocks that make up the symmetric scene known as asymmetric units.

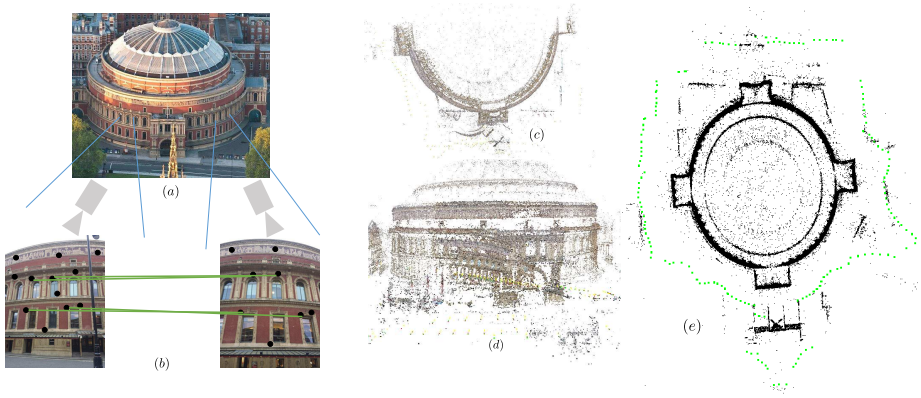


Figure 6.1: **Vanilla implementations of SfM fails under the presence of symmetry**: A duplicate of the image, I showed in the introduction demonstrating the problem assosiated with SfM in the presence of symmetry.

In this chapter my goal is to expand the above algorithm by determining the geometry of a scene from many images of the symmetric structure taken with cameras at different viewing locations. The problem of determining the structure of scene by exploiting cameras at different viewing locations is called **structure from motion** (SfM). Structure from motion in the presence of symmetry is an important problem that is still unsolved.

SfM of a symmetric structure is a hard problem because of the challenge that symmetry poses to the feature matching stage of the pipeline. Feature matching is a critical part of the process of reconstruction from images. Good features are defined based on the ability to reliably detect them across multiple images when the camera has undergone rotations and translations. Feature detection techniques rely on local image statistics that extend across a few pixels and features built this way are not invariant to large camera transformations or to repetitive structures present in the scene. The "Lowe's trick" to match features which is perhaps the most common matching technique used in SfM tends to discard away non-unique features as incorrect matches. While this is indeed useful in

avoiding false positives, it also discards away symmetric matches which do not fall under its bracket of uniqueness.

It is not possible to build descriptors that incorporate a general symmetry information because of the global and diverse nature of symmetry. However, this has been attempted for a select set of symmetries. Not all features use local information exclusively, some techniques use global information such as Koeser *et al.* [KK07] use depth sensors to build a projectively normalized image, Rothganger *et al.* [Rot+06] use approximate camera locations from SfM to match MSER regions. Both these methods require the scene geometry (in the first case, depth and in the second case, camera pose) to build these features, a luxury that is often not available. Some methods successfully solve this problem by specifically tailoring an approach for a certain kind of symmetry. Appearance-based method work well when the symmetry is translational. Wu *et al.* [WFP10; Wu10] proposes a feature called 'Viewpoint Invariant Patch' (VIP) that is invariant to repetition of structures. The key idea in VIP is to use local planar approximations in images to generate frontal views. This transformation removes the invariance to camera transformation and brings all the images into a single view (fronto) on which the other features like SIFT [Low99] can be used for detection and matching. Another recent work that deals with the problem of repetition is by Hauagge *et al.* [HS12] which build features that are invariant to local symmetries by using simple measures of bilateral and rotational symmetries. One of the more recent works on feature matching under the presence of symmetry was done by Loy *et al.* [LE06] who exploits the orientations of SIFT descriptors to extract the symmetry present in the image. By doing so they detect many circular and rotationally symmetric regions in the image.Figure 6.1 shows the output of standard SfM pipeline when the scene

exhibits strong rotational symmetry.

Related works in the area of structure from motion in the presence of symmetry is plentiful [Cey14; Nia12; Coh+12; Rob+11; HS12; Hon+04] but does not adequately address a general class of symmetries. Some of the prominent recent works that jointly perform the task of 3D reconstruction and symmetry detection are restricted to highly specific models [Cey+14], require significant manual intervention [Nia12], are dependent on a good initial reconstruction [Coh+12] or are slow because they recursively update the entire graph and re-optimize for every iteration [SCD15; Coh+12; Cey+14]. The key issue that these techniques suffer from, is the problem of incorrect matches generated due to symmetry. While symmetry provides a global constraint on a macro scale, point features are indistinguishable locally in the presence of symmetry. As a result of this, they are discarded away in the geometric verification step of the pipeline from which it is not possible to recover unless the reconstructed model is approximately correct [Coh+12]. Most importantly, algorithms structured this way have to perform the highly computational task of Bundle Adjustment every time the graph is updated according to the global constraint. Using symmetry invariant features such as line features [SSS09; Cey+14] or point features that are symmetry invariant [HS12; Wu10] can address some of the issues. However, lines detected in images suffer from practical issue like shadows and lack of detail and fail when the assumption of planarity is invalid. While symmetry-aware features [HS12; WFP10] may seem like a solution to this problem, these features only consider a local neighborhood to encode symmetry information [HS12] or they require geometric information from SfM.

In this chapter, I address some of the issues listed above and I demonstrate how one can

determine the generative models described in Chapter 2 in a multi-view, structure from motion setting. In particular, I demonstrate the following improvements over the results obtained in Chapter 4 by exploiting multi-view constraints: (1) obtain denser 3D models comprising of points in non-symmetric regions coexisting with the symmetric ones in a single 3D model (2) Improve geometry of the scene in comparison with traditional SfM methods that do not model symmetry and (3) I obtain even denser and more detailed models of asymmetric units than in the single image case.

I provide results for three specific configurations in the multi-view setting: (1) in the first case, I perform SfM first and then a single manual intervention on one of the images in the SfM set; (2) in the second case, I address the issue when we cannot reliably perform SfM first and I provide a way to fuse the results of the multiple single view reconstructions of the same scene taken from different views and, (3) in the final case, I provide a method to refine the lattice parameters obtained using the voting scheme described in Chapter 4 and demonstrate that this can improve the quality of reconstruction without the need for any manual intervention in the entire pipeline.

SfM without Symmetry

Problem Statement

In order to discuss the problem of SfM in the presence of symmetry, I first briefly introduce the problem (and the related notations) for the case when there is no symmetry. I use

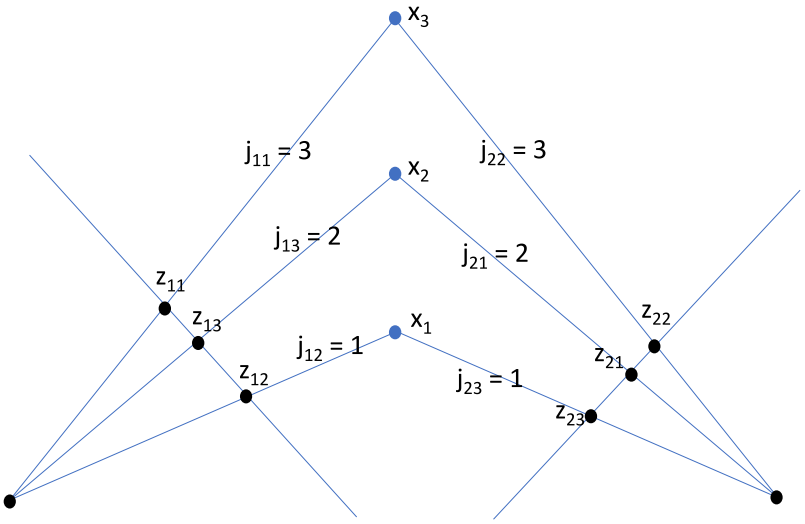


Figure 6.2: **Structure From Motion without Symmetry**: An example problem demonstrating structure from motion of three points $\mathcal{X} = \{\mathsf{x}_1, \mathsf{x}_2, \mathsf{x}_3\}$ from cameras $\mathcal{C} = \{c_1, c_2\}$. The correspondence information in the first image is modeled by the variable $J_1 = \{j_{11} = 3, j_{12} = 2, j_{13} = 1\}$ for each measurement in that image given by $\mathcal{Z} = \{\mathsf{z}_{11}, \mathsf{z}_{12}, \mathsf{z}_{13}\}$. Similarly for the second camera c_2, the correspondence information is given by $J_2 = \{j_{21} = 2, j_{22} = 3, j_{23} = 1\}$ and the measurements is given by $\mathcal{Z} = \{\mathsf{z}_{21}, \mathsf{z}_{22}, \mathsf{z}_{23}\}$.

[Del+00] as reference because I use the correspondence information later on. Consider the situation where a set of n 3D points $\mathcal{X} = \{\mathsf{x}_j\}_{j=0}^{n}$, being observed by $\boldsymbol{c}$ cameras. A measurement z_k is a 2D feature in the i^{th} camera c_i observing the j^{th} point x_j where the correspondence information $k = \mathbf{k}_{ij}$ is known for a total of K measurements (the index k is a function of the camera index i and the point index j, we represent this relationship as $\mathbf{k}_{ij}$). The correspondence information is modeled using the variable $\boldsymbol{j}_{ik}$, which indicates that the 3D point $\mathsf{x}_{j_{ik}}$ generates the measurement z_{ik}. This notation is illustrated shown in Fig 6.2.

In order to optimize for the unknown 3D points and the cameras, we need to specify the measurement function. The measurement function $h\left(\boldsymbol{c}_i, \mathsf{x}_j\right) \mathbb{R}^3 \rightarrow \mathbb{R}^2$ predicts the

measurement z_{ik} as

$$\mathsf{z}_{kj} = h\left(\boldsymbol{c}_i, \mathsf{x}_j\right) + \eta_{\mathsf{z}} \tag{6.1}$$

where η is the measurement noise and $j = \boldsymbol{j}_{ik}$. The most common measurement function used in SfM is the transformation of the 3D point to the local coordinate frame of the camera $\boldsymbol{c}_i$ followed by a projection

$$h\left(\boldsymbol{c}_i, \mathsf{x}_j\right) = \Pi\left({}_w\mathsf{R}_{c_i}^T\left(\mathsf{x}_j - {}_w\mathsf{t}_{c_i}\right)\right) \tag{6.2}$$

where $\left({}_w\mathsf{R}_{c_i}, {}_w\,\mathsf{t}_{c_i}\right) \in \mathbb{SE}\left(3\right)$ is the camera pose.

Now that we have defined our measurement function we can now define the log likelihood which will serve as the objective function whose global maxima corresponds to the optimal values of the unknown cameras $\mathcal{C}$ and 3D points $\mathcal{X}$. The maximum likelihood estimate $(\mathcal{X}^\star, \mathcal{C}^\star)$ is given by

$$\left(\mathcal{X}^\star, \mathcal{C}^\star\right) = \arg\max_{(\mathcal{X},\mathcal{C})} \mathcal{L}\left(\mathcal{X}, \mathcal{C}; \mathcal{Z}, J\right) \tag{6.3}$$

$$-\log \mathcal{L}\left(\mathcal{X}, \mathcal{C}; \mathcal{Z}, J\right) = \frac{1}{2\sigma^2}\sum_{i=1}^{\boldsymbol{c}}\sum_{k=1}^{K_i}\left\|\mathsf{z}_{ik} - h\left(\boldsymbol{c}_i, \mathsf{x}_{\boldsymbol{j}_{ik}}\right)\right\|^2 \tag{6.4}$$

with known correspondence information J.Dellaert *et al.* [Del+00] further show that we can solve the SfM problem as a maximum likelihood estimate without using any known correspondence information J. However, in the subsequent section I will assume that J is given.

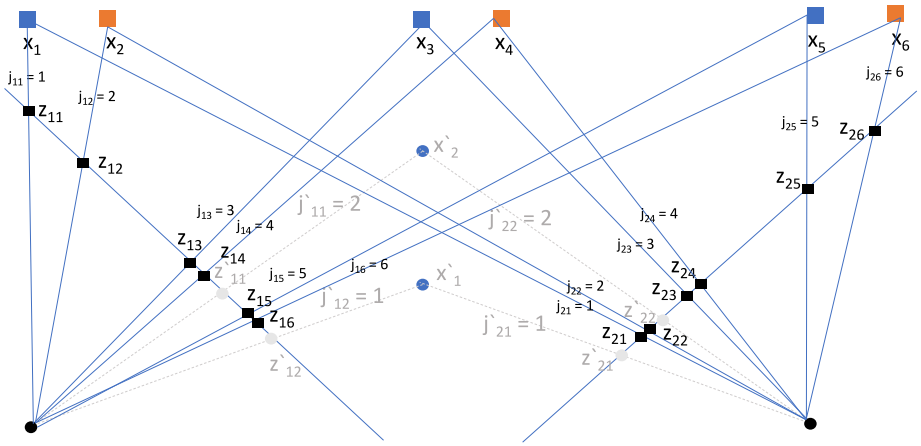


Figure 6.3: **Structure From Motion with Symmetry**: An example problem demonstrating structure from motion of two non-symmetric points $\mathcal{X}\prime = \{\mathsf{x}\prime_1, \mathsf{x}\prime_2\}$ and six symmetric points $\mathcal{X} = \{\mathsf{x}_1 \dots \mathsf{x}_6\}$ and two cameras $\mathcal{C} = \{c_1, c_2\}$. The SfM-only correspondence information (shown in light gray) in the first image is modeled by the variable $J\prime_1 = \{j\prime_{11} = 2, j_{12} = 1\}$ for each measurement in that image given by $\mathcal{Z}\prime = \{\mathsf{z}\prime_{11}, \mathsf{z}\prime_{12}\}$. Similarly for the second camera c_2, the SfM-only correspondence information is given by $J\prime_2 = \{j\prime_{21} = 1, j\prime_{22} = 2\}$ and the measurements is given by $\mathcal{Z} = \{\mathsf{z}\prime_{21}, \mathsf{z}\prime_{22}\}$. There are 6 symmetric points $\mathcal{X}_1 = \{\mathsf{x}_1 \dots \mathsf{x}_6\}$ and the generate 6 measurements in c_1 given by $\mathcal{Z}_1 = (\mathsf{z}_{11}, \mathsf{z}_{12} \dots \mathsf{z}_{16})$. The correspondence information for camera c_1 is modeled by $J_1 = \{j_{11} \dots j_{16}\}$. Similarly the set $\mathcal{X}_2, J_2, \mathcal{Z}_2$

SfM with Symmetry

Problem Statement

I model SfM with symmetry by separating the 3D structure into symmetric and non symmetric sets of points. For the case of SfM in the presence of symmetry, let us consider the following case. The input structure exhibits local symmetry and has two sets of points: (1) $\mathcal{X}' = \{\mathsf{x}\prime_{j\prime}\}_{j\prime=0}^{n\prime-1}$ non-symmetric points and (2) $\mathcal{X} = \{\mathsf{x}_j\}_{j=0}^{n-1}$ points being observed by $\boldsymbol{c}$ cameras. Similarly, there are two sets of measurements corresponding to the two types

of points: (1) A measurement $\mathsf{z}\prime_{k\prime}$ is a 2D feature in the i^{th} camera c_i observing the $j\prime^{th}$ point $\mathsf{x}\prime_{j\prime}$ where the correspondence information $k\prime = \mathbf{k}\prime_{ij\prime}$ is known for a total of $K\prime$ measurements and (2) A measurement z_k is a 2D feature in the i^{th} camera c_i observing the j^{th} symmetric point $\mathsf{x}_{j\prime}$ where the correspondence information $k = \mathbf{k}_{ij}$ is known for a total of K measurements. The correspondence information is modeled by two sets of variables (1) $\boldsymbol{j}\prime_{ik\prime}$, which indicates that the non symmetric 3D point $\mathsf{x}\prime_{j\prime_{ik\prime}}$ generates the measurement $\mathsf{z}\prime_{ik\prime}$. This notation is same as the one illustrated in Fig 6.2 except that all the variables are represented by an additional prime $\prime$ and (2) $\boldsymbol{j}_{ik}$, which indicates that the symmetric 3D point $\mathsf{x}_{j_{ik}}$ generates z_{ik} this notation is illustrated in Fig 1.2 where the symmetric points are represented by the blue and orange squares and the non-symmetric points are represented by the blue circles. The objective function given in Eq 6.4 is now rewritten using primes as

$$-\log \mathcal{L}\left(\mathcal{X}\prime, \mathcal{C}; \mathcal{Z}\prime, J\prime\right) = \frac{1}{2\sigma^2} \sum_{i=1}^{c} \sum_{k\prime=1}^{K\prime_i} \left\| \mathsf{z}\prime_{ik\prime} - h\left(\boldsymbol{c}_i, \mathsf{x}\prime_{\boldsymbol{j}\prime_{ik\prime}}\right)\right\|^2 \tag{6.5}$$

The generative model for the symmetric set of points $\mathcal{X}$ is an extension of the model we introduced in Chapter 2 with indices and correspondences modeled for the multi-view case. In particular, in the measurement function that maps points$\{\mathsf{x}\prime_{ik\prime}, \mathsf{x}_{ik}\}$ to measurements $\{\mathsf{z}\prime_{ik\prime}, \mathsf{z}_{ik}\}$ for each camera c_i, for the case of symmetry, we need to additionally model the generation of the symmetric points $\mathcal{X}$ which preserve the chain of correspondence information from camera to 3D points to asymmetric units. The problem is defined as follows: there are P points $\mathcal{U} = \{\mathsf{u}_p\}_{p=0}^{P-1}$ in the asymmetric unit of a sub-dimensional lattice with depth along the projections in perpendicular directions to the direction of u

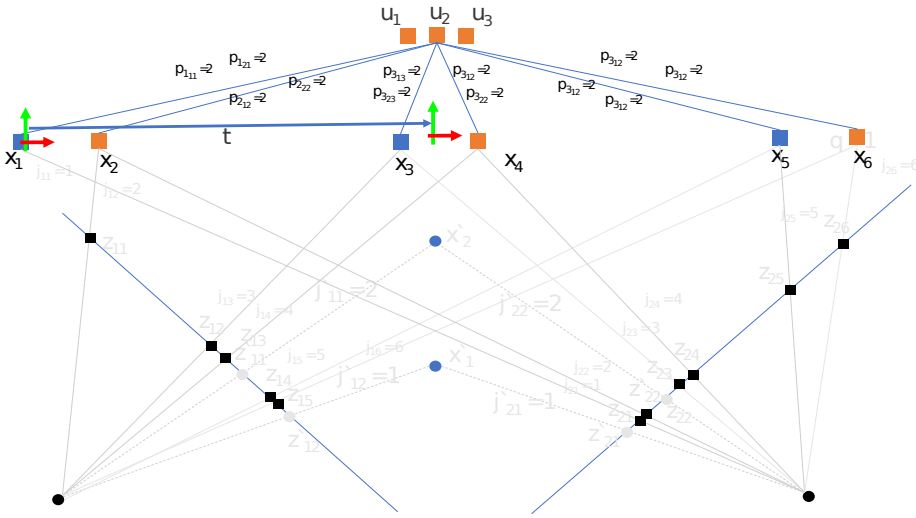


Figure 6.4: **From Asymmetric units to measurements in multiple Images**: The 6 symmetric points $\mathcal{X} = \{\mathsf{x}_1 \dots \mathsf{x}_6\}$. Using the generative model described in Chapters 2&4, we illustrate here that all the points$\mathcal{X}$ are generated by a single point in the asymmetric unit u_2 . Every point in $\mathcal{X}$ has Miller indices $\mathcal{Q} = \{\mathsf{q}_1 \dots \mathsf{q}_6\} \Big| \mathsf{q} \in \mathbb{Z}$ and a Wyckoff vector $\mathcal{W} = \{\mathsf{w}_1 \dots \mathsf{w}_6\} \Big| \mathsf{w} = \{\{a, b\}, \{0, 1\}\}$. The correspondence information in modeled by the variable $\mathbf{p}$ which indexes into the set of points in the asymmetric unit. $\mathsf{u}_{\mathbf{p}_j}$ variable indicates that the j^{th} point in $\mathcal{X}$ corresponds to the $\mathbf{p}_j^{th}$ point in $\mathcal{U}$ and $\mathcal{V}$. We can use the correspondence indicator variable between cameras $\mathcal{C}$ and 3D points $\mathcal{X}$, $\mathbf{j}$ and say that the k^{th} measurement in the i^{th} camera corresponds to the $\mathsf{u}^{th}_{\mathbf{p}_{\mathbf{j}_{ik}}}$ point. The purpose of this figure is to illustrate how a single point in the asymmetric unit generates multiple measurements in the image.

given by $\mathcal{V} = \{\mathsf{v}_p\}_{p=0}^{P-1}$. The measurement function for the symmetric points $\mathcal{X}$ is given by

$$\mathsf{z}_{ik} = h_s\left(c_i, \mathsf{x}_j\right) + \eta_{\mathsf{z}} \tag{6.6}$$

where the point x_j is generated according to a predefined set of symmetry parameters that we know for the points as given in Chapter 2 and 4. The j^{th} symmetric point is generated

by the corresponding p^{th} point in the asymmetric unit as

$$\mathsf{x}_j = h_g\left(\begin{bmatrix} \mathsf{u}_{\mathbf{p}_j} \\ \mathsf{v}_{\mathbf{p}_j} \end{bmatrix}, \Theta, (\mathsf{R}, \mathsf{t})\right) \tag{6.7}$$

where Θ represents all the symmetry parameters of the model. The exact space spanned by this variable is dependent on the kind of symmetry that the scene exhibits which corresponds to a unique generative model that we classified in Chapter 2. A point in the asymmetric unit is represented by $\left(\mathsf{u}_{\mathbf{p}_j} \in \mathbb{R}^L, \mathsf{v}_{\mathbf{p}_j} \in \mathbb{R}^{N-L}\right)$ where $\mathsf{u}_{\mathbf{p}_j}$ represents the projection of the point along the direction of the basis vectors of the lattice and $\mathsf{v}_{\mathbf{p}_j}$ is the projection of the point in directions perpendicular to the basis vectors and also mutually perpendicular to each other.

I now provide a

3D Symmetry, No Lattice

I first define the generative model $h_s(\cdot)$ for the case when there is no lattice. An example of this is the Royal Albert hall dataset that I explored in detail in Chapter 4. For the Royal Albert hall dataset, since the structure can only be partially observed, the unknown symmetry parameters Θ consists of a partially known system of generators g, Wyckoff positions $\mathcal{W}$, and the transformation from the coordinate frame of the asymmetric unit to the global frame, which in this case reduces to a pure rotation as we demonstrated in Chapter 4. In addition to this, the variable $\mathsf{u}_{\mathbf{p}_j} \in \mathbb{R}^L$ does not exist ($L = 0$) and $\mathsf{v}_{\mathbf{p}_j} \in \mathbb{R}^3$.

The generative model for this particular symmetry is given by

$$\begin{aligned} \mathsf{x}_j &= h_g\left(\mathsf{v}_{\mathbf{p}_j} g, \mathsf{w}_j, {}_w\mathsf{R}_a\right) \\ &= \begin{cases} {}_w\mathsf{R}_a\left(\mathsf{g}_1^{\mathsf{w}_{l_p}(1)} \mathsf{g}_2^{\mathsf{w}_j(2)} \mathsf{v}_{\mathbf{p}_j}\right) & \mathsf{w}_j\left(0\right) = d \\ {}_w\mathsf{R}_a\left(\mathsf{g}_2^{\mathsf{w}_j(2)} \mathsf{v}_{\mathbf{p}_j}\right) & \mathsf{w}_j\left(0\right) = c \\ {}_w\mathsf{R}_a\left(\mathsf{g}_2^{\mathsf{w}_j(2)} \mathsf{v}_{\mathbf{p}_j}\right) & \mathsf{w}_j\left(0\right) = b \\ {}_w\mathsf{R}_a\left(\mathsf{v}_{\mathbf{p}_j}\right) & \mathsf{w}_j\left(0\right) = a \end{cases} \end{aligned} \tag{6.8}$$

where ${}_w\mathsf{R}_a \in \mathbb{SO}\left(3\right)$,

$$\mathsf{g}_1 = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$\mathcal{V} = \left\{\mathsf{v}_p\right\}_{p=0}^{P-1} \Big| \mathsf{v}_p \in \mathbb{R}^3$ and $w = \left\{\left\{a, b, c, d\right\}, \left\{0, 1\right\}, \left\{0 \ldots_2 w_N\right\}\right\}$ as explain in detail in the previous Chapter. similar to the single view case, we do not have a method for identifying the special position, we therefore use only the general position as the model for predicting a measurement

$$h_g\left(\mathsf{v}_{\mathbf{p}_j} g, \mathsf{w}_j, {}_w\mathsf{R}_a\right) =_w \mathsf{R}_a\left(\mathsf{g}_1^{\mathsf{w}_{l_p}(1)} \mathsf{g}_2^{\mathsf{w}_j(2)} \mathsf{v}_{\mathbf{p}_j}\right) \tag{6.9}$$

3D Symmetry, 1D Lattice

I now define the generative model for the case of lattices. I provide a specific model that is popular in urban architecture and the one which we discussed in the previous chapter so

that the reader can establish a correspondence quickly between the single and multi view case. The symmetry parameters Θ consists of $g = \{\mathsf{g}_l\}$ the system of 1 generator that is known and models the bilateral symmetry, $\mathcal{W}$, the Wyckoff positions that indicates the position of the point in the asymmetric unit and the number of compositions of each of the generators in g that has to be applied to determine its position inside the unit cell. In addition we have the lattice parameters $(\mathsf{B}, \mathcal{Q})$. The generative model of all datasets that exhibit $'Pm'$ structure is given by

$$
\begin{aligned}
\mathsf{x}_j &= h_g\left(\begin{bmatrix} \mathsf{u}_{\mathbf{p}_j} \\ \mathsf{v}_{\mathbf{p}_j} \end{bmatrix}, \mathsf{q}_j, g, \mathsf{w}_j, \mathsf{B},\right) \\
&= \begin{cases} \mathsf{B}\left(\mathsf{g}_1^{\mathsf{w}_j(1)} \begin{bmatrix} \mathsf{u}_{\mathbf{p}_j} \\ \mathsf{v}_{\mathbf{p}_j} \end{bmatrix} + \begin{bmatrix} \mathsf{q}_j \\ 0 \end{bmatrix}\right) & \mathsf{w}_j\left(0\right) = c \\ \mathsf{B}\left(\mathsf{g}_1^{\mathsf{w}_j(1)} \begin{bmatrix} 0 \\ \mathsf{v}_p \end{bmatrix} + \begin{bmatrix} \mathsf{q}_j \\ 0 \end{bmatrix}\right) & \mathsf{w}_j\left(0\right) = c \\ \mathsf{B}\left(\mathsf{g}_1^{\mathsf{w}_j(1)} \begin{bmatrix} 0 \\ \mathsf{v}_p \end{bmatrix} + \begin{bmatrix} \mathsf{q}_j \\ 0 \end{bmatrix}\right) & \mathsf{w}_j\left(0\right) = b \end{cases}
\end{aligned} \tag{6.10}
$$

where

$$
\mathsf{B} = \begin{bmatrix} b & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}
$$

$$\mathsf{g}_1 = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$\mathcal{U} = \{\mathsf{u}_p\}_{p=0}^{P-1} \left| \mathsf{u}_p \in \mathbb{R}^1, \mathcal{V} = \{\mathsf{v}_p\}_{p=0}^{P-1} \right| \mathsf{v}_p \in \mathbb{R}^2$ and $\mathsf{q}_j \in \mathbb{Z}, w = \{\{a, b, c\}, \{0, 1\}\}$. There is no relative transformation modeled here because, without the loss of any generality, we can choose the translation of the asymmetric unit at the 0^{th} location of the Miller Indices and the rotation as idenity as described in Chapter 5.

To complete the story, we also have to describe how the generated 3D points are projected onto multiple images. Eq 6.6 predicts the measurement z_{ik} of a point in the asymmetric unit $\begin{bmatrix} \mathsf{u}_{\mathbf{p}_j} \\ \mathsf{v}_{\mathbf{p}_j} \end{bmatrix}$ observed by camera c_i where $h_s(\cdot)$

$$h_s(c_i, \mathsf{x}_{\mathbf{j}_{ik}}) = \Pi \left({}_w\mathsf{R}_{c_i}^T \left(h_g \left(\begin{bmatrix} \mathsf{u}_{\mathbf{p}_j} \\ \mathsf{v}_{\mathbf{p}_j} \end{bmatrix}, \mathsf{q}_j, g, \mathsf{w}_j, \mathsf{B}, \right) - {}_w\,\mathsf{t}_{c_i} \right) \right) \tag{6.11}$$

$$h_s(c_i, \mathsf{x}_{\mathbf{j}_{ik}}) = \Pi \left({}_w\mathsf{R}_{c_i}^T \left(h_g \left(\mathsf{v}_{\mathbf{p}_j} g, \mathsf{w}_j, {}_w\,\mathsf{R}_a \right) - {}_{c_i}\,\mathsf{t}_w \right) \right) \tag{6.12}$$

Using the measurement functions described above we can now optimize in for the location for symmetric structure comprising of points in the asymmetric unit, the parameters of lattice if it exists and the system of generators if unknown together with the location of multiple cameras and non-symmetric points $\mathcal{X}\prime$. The optimal value of these parameters is a maximum likelihood estimate of $(\mathcal{X}^\star, \mathcal{C}^\star, \mathcal{U}^\star, \mathcal{V}^\star, \Theta^\star)$ considering the symmetric

measurements only, we get

$$(\mathcal{C}^\star, \mathcal{U}^\star, \mathcal{V}^\star, \Theta^\star) = \max_{(\mathcal{C},\mathcal{U},\mathcal{V},\Theta)} \mathcal{L}\left(\mathcal{C}, \mathcal{U}, \mathcal{V}, \Theta; \mathcal{Z}, J, \mathsf{P}, \mathcal{Q}\right) \tag{6.13}$$

however, we also have measurements from non-symmetric regions, incorporating this into the energy function we get the total maximum likelihood of $(\mathcal{X}\prime^\star, \mathcal{C}^\star, \mathcal{U}^\star, \mathcal{V}^\star, \mathsf{B}^\star, {}_w\,\mathsf{T}_a^\star, g^\star)$

$$(\mathcal{X}\prime^\star, \mathcal{C}^\star, \mathcal{U}^\star, \mathcal{V}^\star, \Theta^\star) = \max_{(\mathcal{X}\prime,\mathcal{C},\mathcal{U},\mathcal{V},\mathsf{B},{}_w\mathsf{T}_a,g)} \mathcal{L}\left(\mathcal{X}\prime, \mathcal{C}, \mathcal{U}, \mathcal{V}, \Theta; \mathcal{Z}, J, \mathsf{P}, \mathcal{Q}\right) \tag{6.14}$$

where

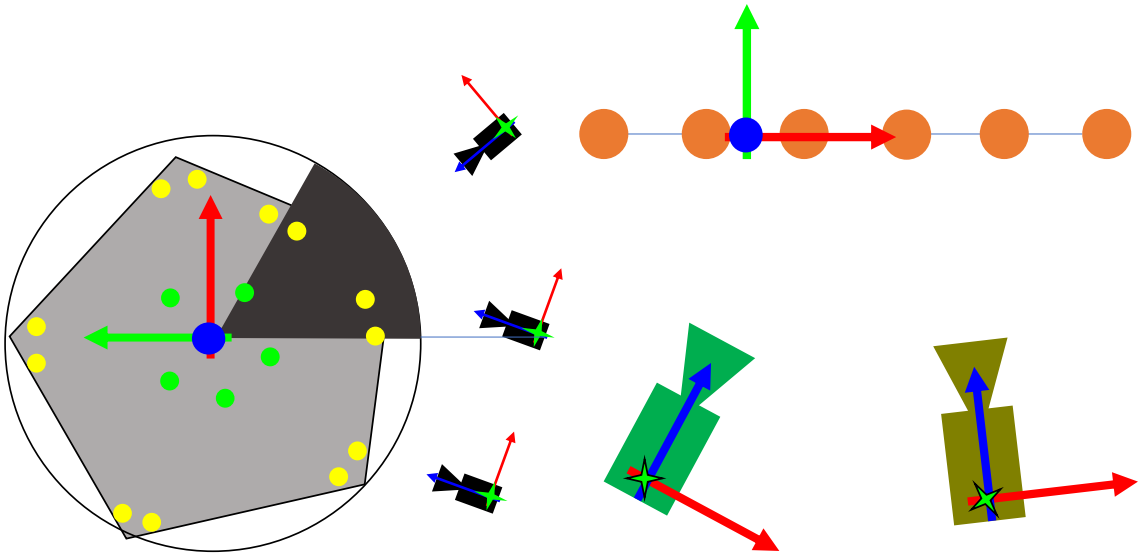
$$\begin{aligned} -\log \mathcal{L}\left(\mathcal{X}\prime, \mathcal{C}, \mathcal{U}, \mathcal{V}, \Theta; \mathcal{Z}, J, \mathsf{P}, \mathcal{Q}\right) &= \frac{1}{2\sigma^2} \sum_{i=1}^{\boldsymbol{c}} \sum_{k\prime=1}^{K\prime_i} \left\| \mathsf{z}\boldsymbol{\prime}_{ik\prime} - h\left(\boldsymbol{c}_i, \mathsf{x}\boldsymbol{\prime}_{\boldsymbol{j}\prime_{ik\prime}}\right)\right\|^2 \\ &+ \frac{1}{2\sigma^2} \sum_{i=1}^{\boldsymbol{c}} \sum_{k=1}^{K_i} \left\| \mathsf{z}_{ik} - h_g\left(\boldsymbol{c}_i, \mathsf{x}_{\mathbf{j}_{ik}}\right)\right\|^2 \end{aligned} \tag{6.15}$$

where P is a variable that models the correspondence between point in the asymmetric unit u_p and the 3D point x_j.

Joint SfM and Symmetry Estimation

We describe 3 cases of multi view reconstruction in the presence of symmetry

1. Case1: refine the full generative model of symmetry described above when we are allowed one manual intervention and can obtain a reconstruction of the scene despite the presence of symmetry.



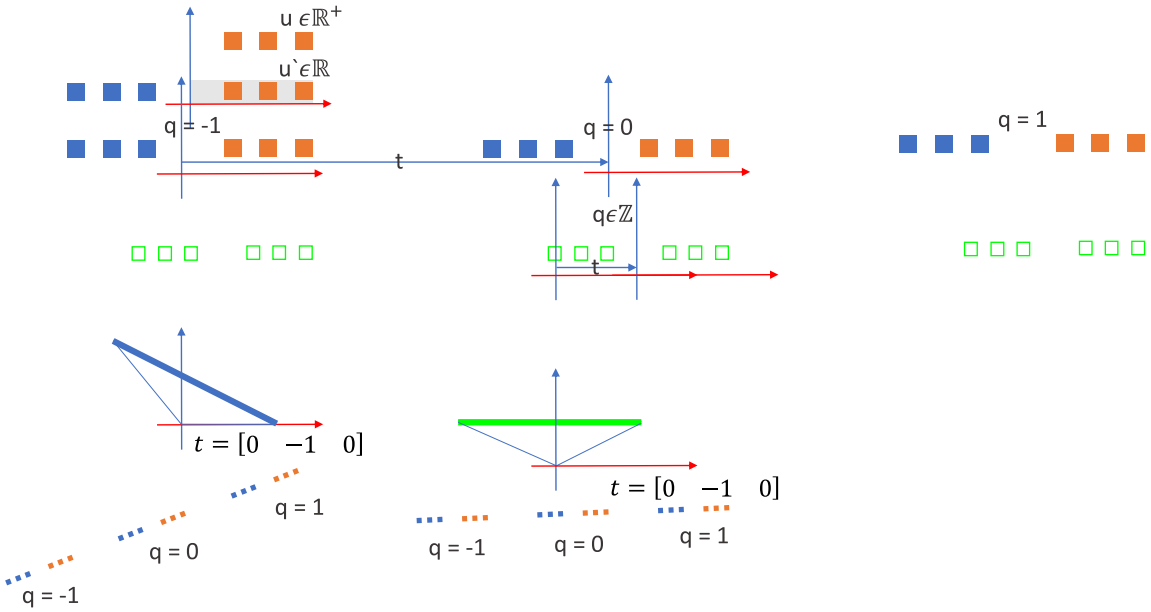
(a) Rotational Symmetry, No lattice (b) 1D Lattice, 3D symmetry

Figure 6.5: **Gauge Freedom Multi-view**: The gauge freedom associated with multi-view symmetry estimation is limited only by the total Gauge freedom of the structure from motion and it is agnostic to the symmetry of the scene unlike the single image case.

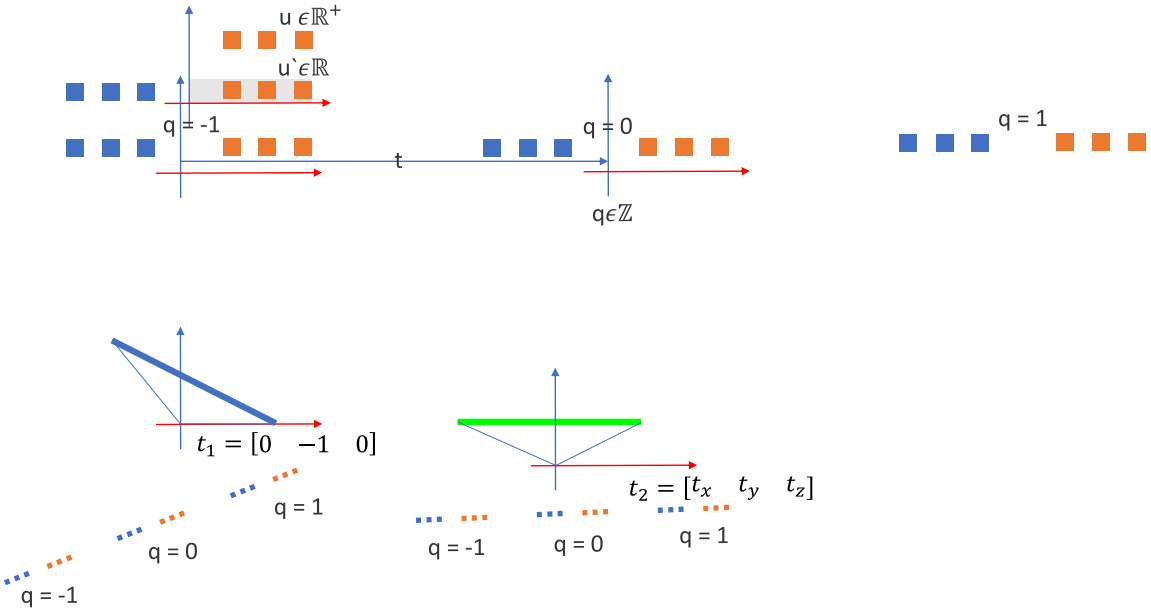
2. Case2: obtain a plausible model of the scene when the images suffer from an aliasing effect, as a result of which we cannot perform traditional structure from motion as required in Case 1.

3. Case3: refine lattice parameters obtained from voting on a noisy SfM cloud by exploiting multi-view camera constraints.

I start with the first case below:



(a) Problem with Multiple Manual Intervention:



(b) Solution using SfM and Guided Matching

Figure 6.6: **Case 1: Joint estimation of Asymmetric unit and the Camera geometry:** (a) Here, we demonstrate the problem of fusing multiple, single image reconstructions. As we saw in Chapter 4, every camera is initialized at $(0, -1, 0)$ and as result the structure has different scales and orientations. (b) In order to jointly optimize for the pose of both cameras and the structure we need a globally consistent assignment of Miller indices $\mathcal{Q}$ and measurements in each camera to correspond to the correct point in the asymmetric unit.

Case1 : Refine Generative Model of Symmetry after SfM

The assumptions we make for this case are as follows: (1) Given a set of images corresponding to $\boldsymbol{c}$ cameras, we can successfully perform traditional structure from motion where the relative geometry of the scene and the cameras is not too distorted. This is contingent upon having strong non-symmetric regions in the images that can serve as anchors to determine the relative geometry of the scene (2) We can perform a single manual intervention on one of the images from our input set, although, choosing an image which observes majority of the symmetric part of scene is beneficial but not necessary to the success of the algorithm.

Problem Statement

In Figure 6.6 I show the major problem that we have to solve and the solution for the current case. In Figure 6.6(a), I show the issue of fusing reconstructions from many single views using the technique outlined in Chapter 4. In Figure 6.6(b) I show the variables that need to be determined, namely the miller Indices $\mathcal{Q}$, the correspondence information between cameras and measurements in multiple views, J and the symmetry correspondence information P that models the indices of the point in the asymmetric unit and 3D points $\mathcal{X}$. Given this information, we can jointly optimize for the camera poses points in asymmetric unit in a multi-view setting.

The objective function for this scenario is exactly same as that defined in Section 6.3. Since, it has both symmetric and non-symmetric points, the objective function under the

assumption of known multi-view symmetry correspondence information is given by

$$
\begin{aligned}
-\log \mathcal{L}\left(\mathcal{X}\prime, \mathcal{C}, \mathcal{U}, \mathcal{V}, \mathsf{B}, {}_{w}\mathsf{T}_{a}, g; \mathcal{Z}, J, \mathsf{P}, \mathcal{Q}\right) &= \frac{1}{2\sigma^2}\sum_{i=1}^{\mathbf{c}}\sum_{k\prime=1}^{K\prime_i}\left\|\mathsf{z}\prime_{ik\prime} - h\left(\boldsymbol{c}_i, \mathsf{x}\prime_{\mathbf{j}\prime_{ik\prime}}\right)\right\|^2 \\
&+ \frac{1}{2\sigma^2}\sum_{i=1}^{\mathbf{c}}\sum_{k=1}^{K_i}\left\|\mathsf{z}_{ik} - h_s\left(\boldsymbol{c}_i, \mathsf{x}_{\mathbf{j}_{ik}}\right)\right\|^2 \quad (6.16)
\end{aligned}
$$

Guided Matching

In this section I describe a guided matching scheme that can establish the Miller Indices $\mathcal{Q}$ and the correspondence variables P and J, which can allow us to use the optimization scheme described in Section 6.5 to determine the cameras $\mathcal{C}$, non symmetric points $\mathcal{X}\prime$ and the points in the asymmetric unit $(\mathcal{U}, \mathcal{V})$ along with the symmetry parameters $(\mathsf{B}, {}_{w}\mathsf{T}_{a}, \mathsf{g})$ in a multi-view setting. From Equation 6.16, we can see that for every point $\mathsf{x}_{\mathbf{j}_{ik}}$, we need its Wyckoff variable $\mathsf{w}_{\mathbf{j}_{ik}}$ and corresponding point in the asymmetric unit $\left(\mathsf{u}_{p_{\mathbf{j}_{ik}}}, \mathsf{v}_{p_{\mathbf{j}_{ik}}}\right)$.

I first perform a global structure from motion comprising of all the images to obtain the relative camera geometry (shown in Figure 6.7(a)) and a single view reconstruction on one of the images(shown in Figure 6.7(b)) to determine the correspondence variables P and J. I outline the steps for the multi-view guided matching algorithm as follows:

1. Given we know the set of cameras $\mathcal{C}$ from SfM, I first perform single image reconstruction using the method outlined in Chapter 4 by fixing the camera pose and calibration and optimizing for the points in the asymmetric unit and the symmetry parameters Θ.

2. I then perform guided matching within the image as detailed in Section 5.5 to increase the density of the points (and therefore, the matches) within a single image.

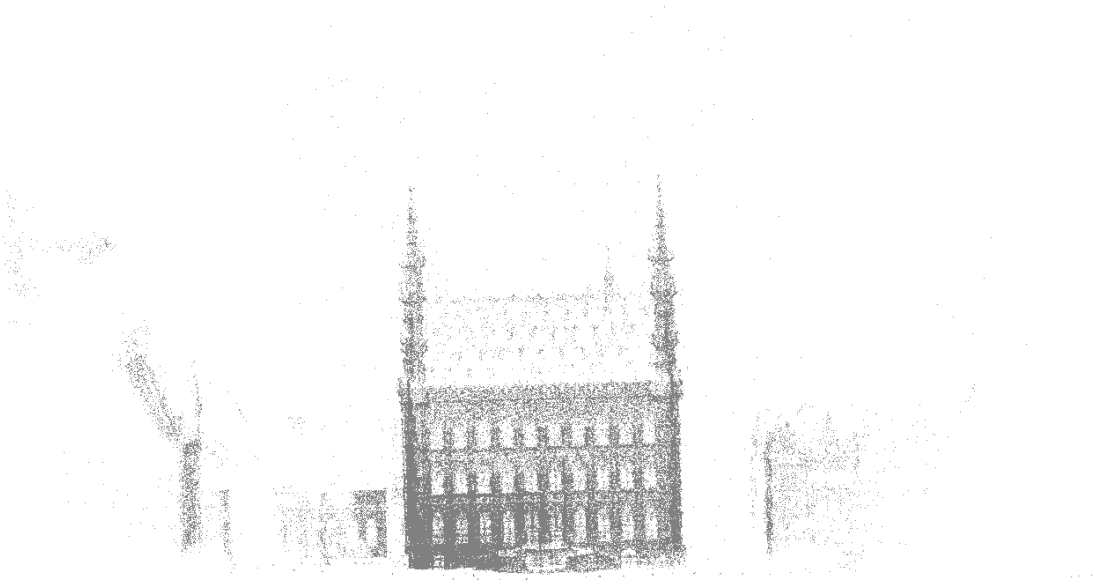
3. I then generate all the symmetric 3D points $\mathcal{X}$ by using the full generative model of symmetry that is determined for this particular view. It is worth mentioning that it is beneficial to pick an image that has full information on the symmetric structure expanding the range of our observation to as many images as possible in the multi-view setting.

4. Given the set of points $\mathcal{X}$ and its associated measurements $\mathcal{Z}$ obtained from the single view case, I expand the set of measurements $\mathcal{Z}$ to multiple views by using the relative geometry of the camera and the appearance of the scene:

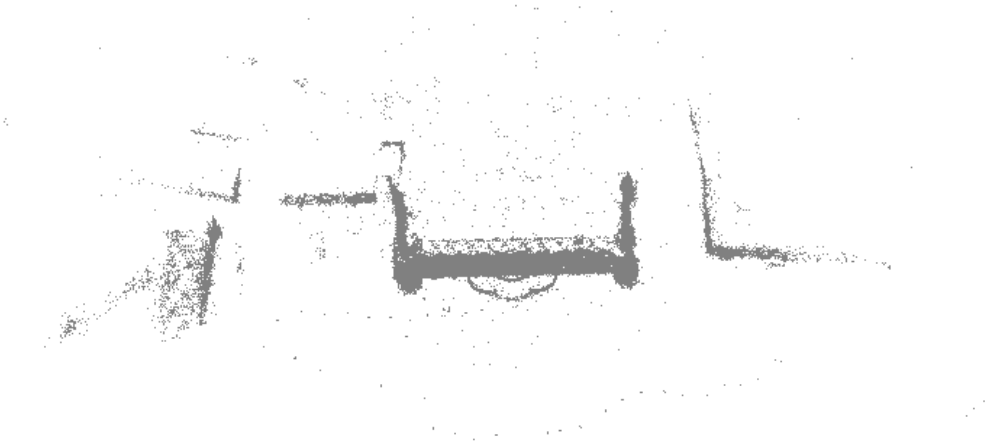
   (a) For every point in $\mathsf{x}_j \in \mathcal{X}$, I project the points on all the cameras in the camera set $c_i \in \mathcal{C}$(except our current camera) using Equation 6.2.

   (b) If the projected location is within the image, I obtain the best match between the set of features centered around the projected location in c_i, that lie within a circle of radius r and the feature corresponding to the point x_j (used in single view reconstruction as a measurement).

   (c) If there is a match, I then update the set of measurements $\mathcal{Z}$ to include this new entry $\mathsf{z}_{\mathbf{k}_{ij}}$ that is generated from the point $\left(\mathsf{u}_{\mathbf{p}_j}, \mathsf{v}_{\mathbf{p}_j}\right)$. In addition, we also update the symmetry and lattice parameters that correspond to this measurement as exactly same as that assigned for x_j during single view reconstruction.



(a) Front View



(b) Top View

Figure 6.7: **Input, Case1:** I show the output of the vanilla SfM motion on the Leuven dataset. This is used to initially obtain the approximate scene geometry, which can later help us with the guided matching to multiple views(Section 4.5.1). Notice the presence of non-symmetric regions around the building that anchor the camera poses even under the presence of symmetry.

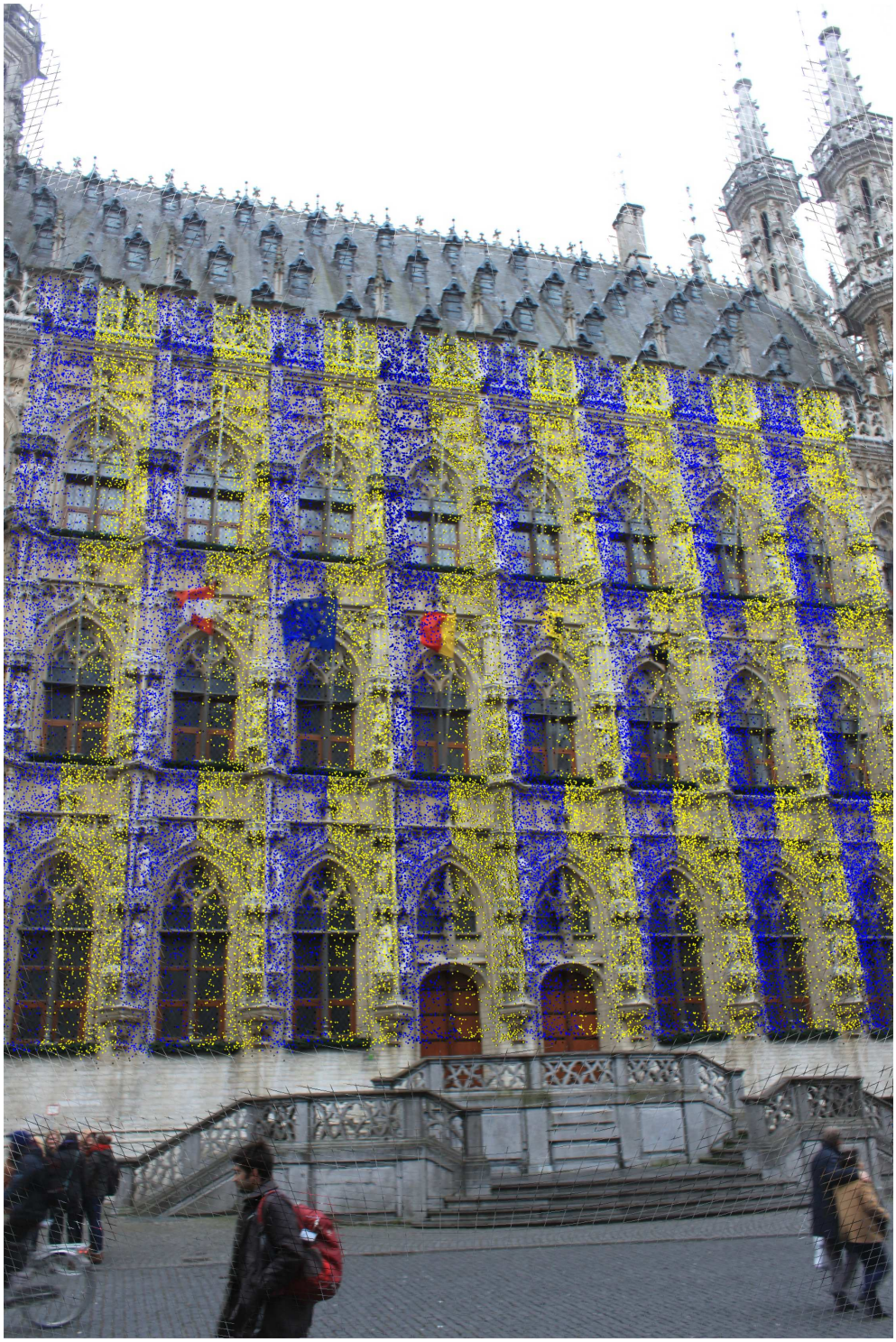
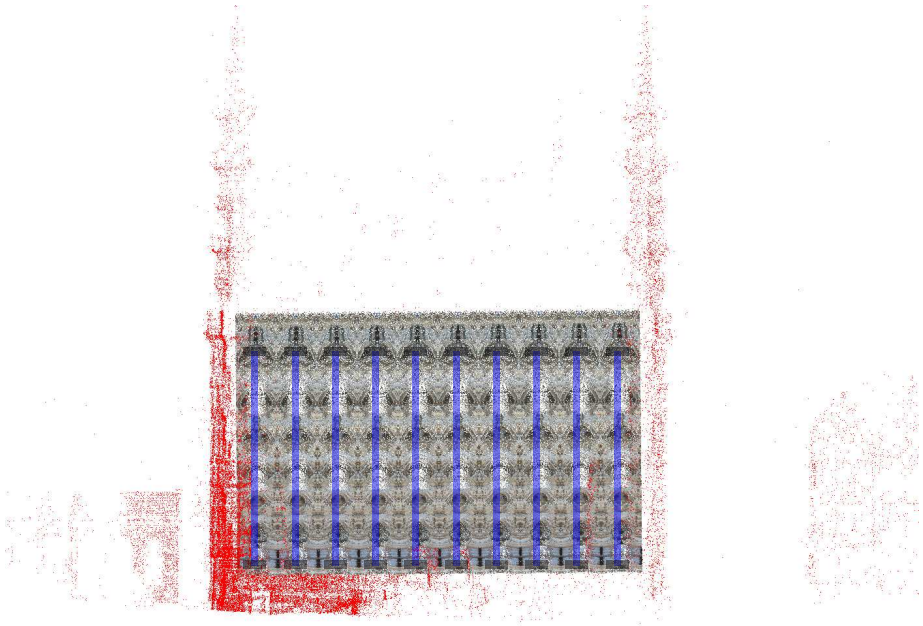


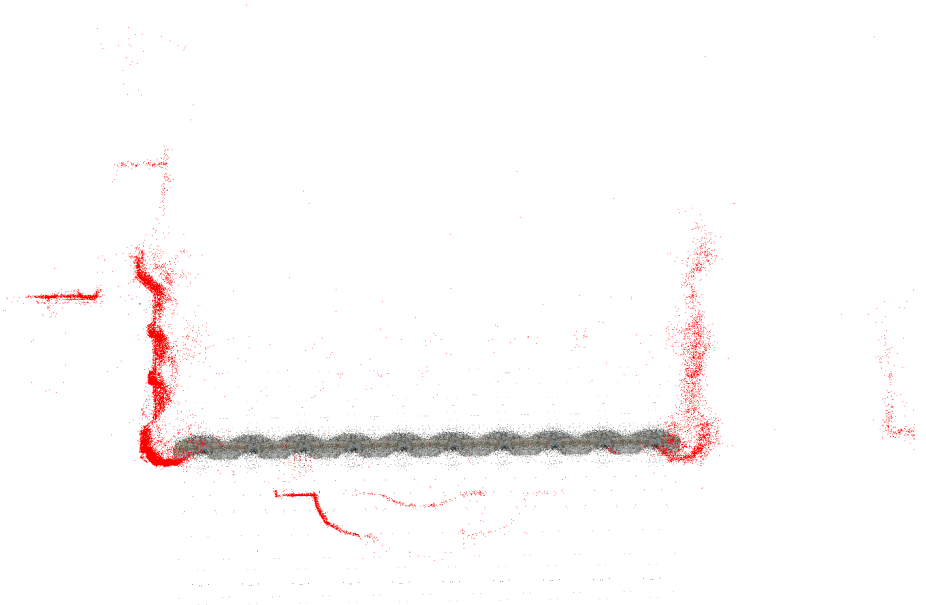
Figure 6.8: **Input, Case1:** Pick an image from set of images used for reconstruction shown in Figure 6.7 and then perform single image reconstruction on this image. The measurements for the points shown in this figure is updated to multiple views using the guided matching scheme discussed in Section 6.6.1.

Results

Figure 6.9 shows the results of multi-view reconstruction under symmetry for the Leuven dataset. The symmetric regions are much denser as compared to the points obtained from SfM alone, thanks to the manual intervention and the guided matching stage in single and multiple views (which essentially multiplies the points and measurements). In fact, the more symmetric regions we identify in the structure, the denser the obtained 3D model (our generative model can in fact model relative transformation between multiple types of symmetries if present in the same structure).



(a) Front view



(b) Top view

Figure 6.9: **Results, Case 1: Joint Structure From motion and Symmetry Estimation**: Here, I show the results for the first case, i.e, when we can obtain the scene geometry from SfM due to the presence of strong non-symmetric regions in the scene. The building adjacent to the Leuven Stadius is especially beneficial for this, also, there are a few feature-rich non symmetric regions in the building itself. I perform joint SfM and symmetry estimation by optimizing the objective function given in Equation 6.16. The points shown in red correspond the non-symmetric optimized points $\mathcal{X}'$ and the the textured points correspond to the symmetric points $\mathcal{X}$.

Case2 : Fusing Multiple Single-view Reconstructions

In the second case I solve the problem of joint structure from motion and asymmetric unit estimation on a scene that is exclusively imaging a symmetric region with **no non-symmetric regions** to estimate the camera poses initially. I solve this problem by performing single image reconstruction on all images individually and assigning a locally consistent but globally arbitrary assignment of correspondence information and extract a scene geometry that is consistent with this assignment.

In Figure 6.10 I show the problem configuration for the current case. I show that the relative camera locations are unknown and I make a random assignment of Miller Indices to measurements in each camera. However, we cannot assign a completely arbitrary assignment in each camera. The indices have to be consistent locally in each camera as we described in the single image reconstruction case. However, we can choose any arbitrary offset between the Miller indices of the the two cameras. In the example problem shown in Figure 6.10, I choose an offset of 3. The true scene geometry of the corresponding scene was shown in Figure 6.6(a).

The objective function for this scenario consists of the symmetric points and the optimization is essentially a subset of the problem we defined in Case1:

$$-\log \mathcal{L}\left(\mathcal{C},\mathcal{U},\mathcal{V},\mathsf{B},{}_{w}\mathsf{T}_{a},g;\mathcal{Z},J,\mathsf{P},\mathcal{Q}\right)=\frac{1}{2\sigma^{2}}\sum_{i=1}^{\mathbf{c}}\sum_{k=1}^{K_{i}}\left\|\mathsf{z}_{ik}-h_{s}\left(\boldsymbol{c}_{i},\mathsf{x}_{\mathbf{j}_{ik}}\right)\right\|^{2} \qquad (6.17)$$

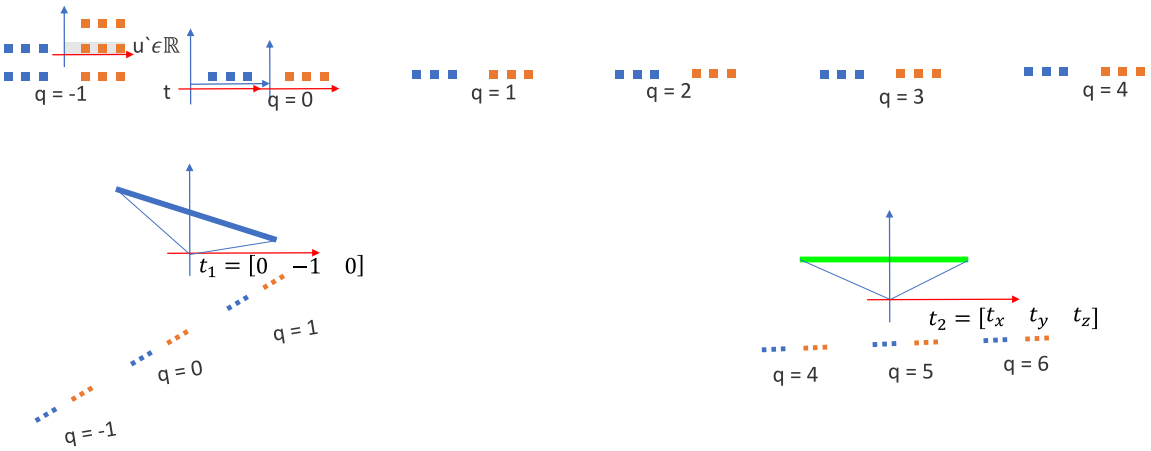
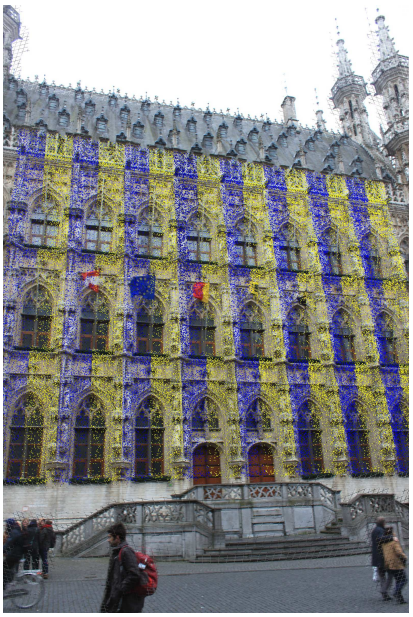
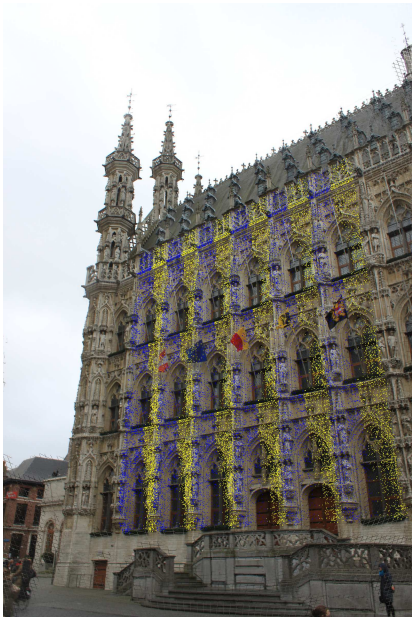
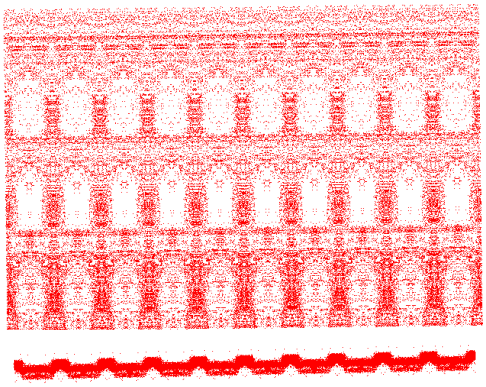


Figure 6.10: **Symmetric Structure from Motion under Aliased views**: Consider two cameras imaging a symmetric scene that is represented by an asymmetric unit. If we perform single image reconstruction separately on both these cameras, this will lead to two inconsistent sets of reconstructions and symmetries $(\mathcal{U}_1, \mathcal{V}_{1,w}\, \mathsf{T}_{a_1}, \mathsf{B}_1 \ldots)$ for camera c_1 and $(\mathcal{U}_2, \mathcal{V}_{2,w}\, \mathsf{T}_{a_2}, \mathsf{B}_2 \ldots)$ for camera c_2. This is graphically shown in Figure 6.6 (a). However, the correspondence information is locally consistent and in this section we describe how much of the correspondence information can we establish by using appearance alone and how we can get a globally consistent geometry by assigning arbitrary values to other correspondence information that we cannot obtain.







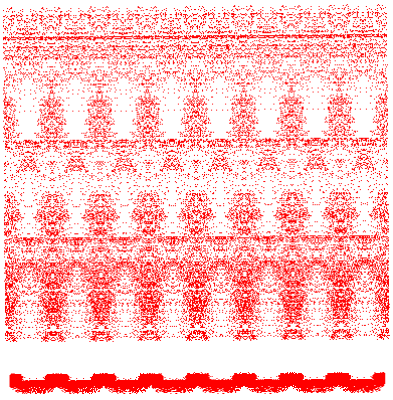


Figure 6.11: **Results, Case2**: In the top row, I show the two images used along with the reconstructed points using single view reconstruction; This the input to our algorithm. We now use the guided matching scheme described in Section 6.6.1 to obtain the multi-view symmetry correspondence information.

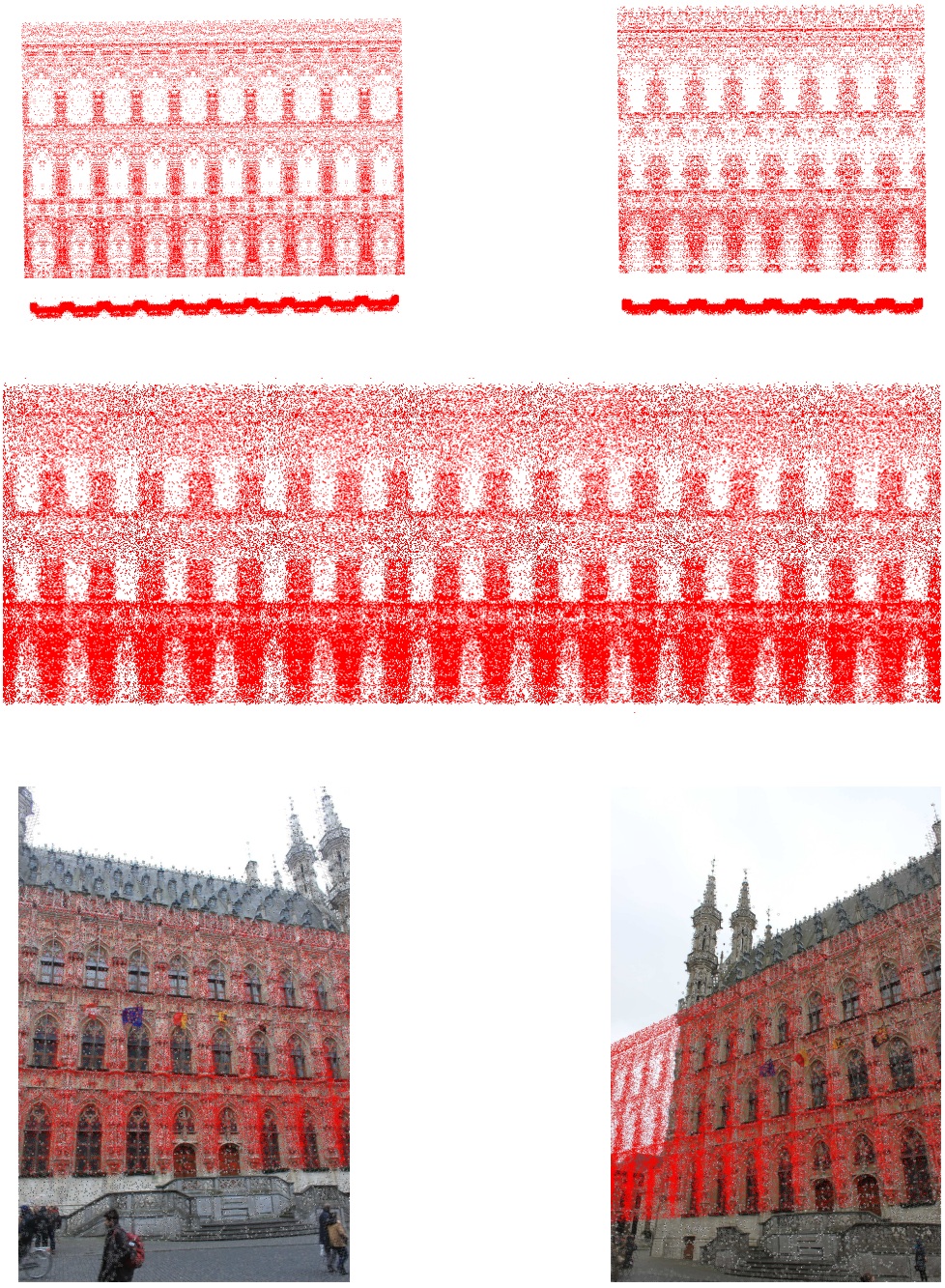


Figure 6.12: **Results, Case2**: In this figure, I show the results of fusing two single image reconstruction after assigning locally consistent and globally arbitrary Miller indices and obtain the corresponding plausible model for this assignment. In the top row, I show the two single image reconstructions that are obtained. In the middle row, I show the plausible model that we obtained by assigning a large separation of Miller Indices between the two cameras. The number of repetitions in the symmetric structure in much greater than is actually present. We project these points onto the two cameras to verify the camera geometry and we can see here that the structure is essentially extending to infinity in the bottom left image.

Guided Matching

Despite the apparently arbitrary assignment of symmetry correspondence information in this problem, we still need to establish consistent matches across multiple views to find the optimum value of the variables given in the objective function in Equation 6.17 in a globally consistent manner.

In this section, I describe an algorithm to obtain matches across 2 views each of which has been reconstructed using the algorithm described in the previous chapter. I once again, for the lack of a better term, describe a guided matching scheme that can establish the Miller Indices $\mathcal{Q}$ and the correspondence variables $\mathsf{P}.J$ which can allow us to use the optimization scheme described in Section 6.5 to determine the cameras $\mathcal{C}$, non symmetric points $\mathcal{X}\prime$ and the points in the asymmetric unit $(\mathcal{U}, \mathcal{V})$ and the symmetry parameters $(\mathsf{B}, _w \mathsf{T}_a, \mathsf{g})$ in a multi-view setting. From Equation 6.16, we can see that for every point $\mathsf{x}_{\mathbf{j}_{ik}}$, we need its Wyckoff variable $\mathsf{w}_{\mathbf{j}_{ik}}$ and corresponding point in the asymmetric unit $\left(\mathsf{u}_{p_{\mathbf{j}_{ik}}}, \mathsf{v}_{p_{\mathbf{j}_{ik}}}\right)$. In order to determine these assignments, we follow these steps:

1. Given two cameras c_1, c_2 imaging the same scene, I first perform single image reconstruction using the method outlined in Chapter 4 to obtain two sets of points $\mathcal{X}_1, \mathcal{X}_2$ and their assosiated symmetry and lattice parameters.

2. For every point in the asymmetric unit $(\mathcal{U}_1, \mathcal{V}_1)$ that generate a subset of points in $\mathcal{X}_1$, I compute the best match (if exists) to a point in the asymmetric unit in the set $(\mathcal{U}_2, \mathcal{V}_2)$ as follows:

(a) for the set of features (called **track**) $\mathcal{X}_1^1$ that correspond to a point in the asymmetric unit $\mathsf{u}_1, \mathsf{v}_1$ observed by camera c_i, I compute all the matches with the set of features in $\mathcal{X}_2$ using approximate nearest neighbor matching scheme.

(b) If there is a match between **all** the points in a track in $\mathcal{X}_1^1$ and all the points in any single track in $\mathcal{X}_2$, I conclude that these tracks correspond to the same point in the asymmetric unit. If there is no match, I only add the measurements from a single image into the measurement set $\mathcal{Z}$.

Case3 : Refinement of Lattice using SfM

Finally, I demonstrate how we can optimize for lattice parameters and 3D geometry simultaneously by using initial estimate of the voting scheme without any manual intervention. I demonstrated in Chapter 3 that we can decouple the estimation of lattices from the estimation of other point groups by treating lattices as a subgroup of the symmetry group of the scene. By doing so, I successfully extracted sub-dimensional lattices from noisy SfM clouds **without** any manual intervention. I further demonstrated that it is more effective to vote in a polar transformation space as votes that are cast in this space are robust to the noisy nature of SfM clouds. In this section, I demonstrate that we can refine the estimated lattice parameters and the scene geometry by solving a sub-problem of that described in Section 6.3. This is an important step because it shows that we can extract and refine lattice without the need for any manual intervention. Therefore, without much ado, I will proceed with defining this sub-problem

Problem Definition

A particular model of symmetry as defined by our generative model, uniquely determines a lattice but the converse is not true. Let us briefly revisit the definition of lattices a group of the discrete translational symmetry and how this viewpoint modifies the generative model described in Section 6.3. Every point in $\mathcal{X}$ is generated from its corresponding point in the asymmetric unit $(\mathcal{U}, \mathcal{V})$ by a two stage process **(1)** by first transforming the point by combination of the system of generators $g_p = \{{}_p\mathsf{g}_1 \ldots {}_p\,\mathsf{g}_P\}$ that define the symmetry operations within the unit cell and (2) translating the transformed point by a vector specified by the basis of the lattice B, its pose in 3D space ${}_w\mathsf{T}_a$, and the Miller indices $\mathcal{Q}$. We showed in Chapter 3 that the lattice specified by the tuple $(\mathsf{B}, {}_w\,\mathsf{T}_a)$ has a one-to-one correspondence with a generator system $g_l = \{{}_l\mathsf{g}_1 \ldots {}_l\,\mathsf{g}_L\}$. Consider two N dimensional points $\mathsf{x}_1, \mathsf{x}_2 \in \mathbb{R}^N$ in a L dimensional lattice defined by a basis B and a transformation ${}_w\mathsf{T}_a$ with respect to a global coordinate frame, having Miller indices of q_1 and q_2, then the relative location between $\mathsf{x}_1, \mathsf{x}_2$, specified by a transformation ${}_1\mathsf{T}_2$ can be expressed as a composition of L **unique generators** $\{{}_l\mathsf{g}_1 \ldots {}_l\,\mathsf{g}_L\}$ such that

$$ {}_1\mathsf{T}_2 = {}_l\mathsf{g}_1^{(\mathsf{q}_1(0)-\mathsf{q}_2(0))} \ldots {}_l\mathsf{g}_L^{(\mathsf{q}_1(L-1)-\mathsf{q}_2(L-1))} \tag{6.18} $$

Conversely, we showed in Chapter 3 that, given a point $\mathsf{x} \in \mathbb{R}^N$ in a L dimensional lattice is defined by a set of generators as $(\{{}_l\mathsf{g}_1 \ldots {}_l\,\mathsf{g}_L\})$. We can map from one representation to the other by using an RQ decomposition of the stacked generators as

$$ \mathsf{R}\,\mathsf{B} = \mathbf{RQ}\left(\begin{bmatrix} {}_l\mathsf{g}_1 & \ldots & \ldots {}_l\,\mathsf{g}_L & 0 \end{bmatrix}\right) \tag{6.19} $$

where $\mathbf{RQ}(\cdot)$ is the RQ decomposition which gives us the rotation of the lattice in $\mathbb{R}^N$ and the basis matrix $\mathsf{B} \in \mathbb{R}^{N \times N}$as an upper triangular matrix. Now let us define the SfM problem.

Consider n_s symmetric points $\{\mathsf{x}_1 \dots \mathsf{x}_{n_s}\} \in \mathcal{X}$ and $n_s\prime$ non symmetric points $\{\mathsf{x}\prime_1 \dots \mathsf{x}\prime_{n\prime_s}\} \in \mathcal{X}\prime$ in a lattice being observed by $\boldsymbol{c}$ cameras. The energy function for the non symmetric points alone is given by

$$-\log \mathcal{L}\left(\mathcal{X}\prime, \mathcal{C}; \mathcal{Z}\prime, J\prime\right) = \frac{1}{2\sigma^2} \sum_{i=1}^{\boldsymbol{c}} \sum_{k\prime=1}^{K\prime_i} \left\| \mathsf{z}\prime_{ik\prime} - h\left(\boldsymbol{c}_i, \mathsf{x}\prime_{j\prime_{ik\prime}}\right) \right\|^2 \tag{6.20}$$

In addition to this every symmetric point x_j generates virtual measurements in cameras that would otherwise not be able to see this points by means of the lattice relation:

$$\mathsf{z}_{ik} = h_l\left(c_i, \mathsf{x}_j, \mathsf{q}_j, g_l\right) + \eta_{\mathsf{z}}$$

even though camera c_i does not directly observe point x_j. The measurement function $h_l(\cdot)$

$$h_l\left(c_i, \mathsf{x}_j, g_l; \mathcal{Q}\right) = \Pi\left({}_w\mathsf{R}_{c_i}\left({}_l\mathsf{g}_1^{\mathsf{q}_{j1}} \circ {}_l\mathsf{g}_2^{\mathsf{q}_{j2}} \circ \dots {}_l\mathsf{g}_L^{\mathsf{q}_{jL}}\right)\mathsf{x}_j - {}_w\,\mathsf{t}_{c_i}\right) \tag{6.21}$$

where $\mathsf{q}_j = [\mathsf{q}_{j1} \dots \mathsf{q}_{jL}]$ is the known Miller indices of the translated point assuming that x_j is the origin. $({}_w\mathsf{R}_{c_i}, {}_w\,\mathsf{t}_{c_i})$ is the camera pose. The energy function corresponding to the symmetric points in the lattice is given by

$$-\log \mathcal{L}\left(\mathcal{X}, \mathcal{C}, g_l; \mathcal{Z}, J, \mathcal{Q}\right) = \frac{1}{2\sigma^2} \sum_{i=1}^{\boldsymbol{c}} \sum_{k=1}^{K_i} \left\| \mathsf{z}_{ik} - h_l\left(\boldsymbol{c}_i, \mathsf{x}_{\mathbf{j}_{ik}}, g_l\right) \right\|^2 \tag{6.22}$$

where K_i includes both the real and the virtual measurements observed by c_i. The total energy function is then given by

$$\begin{aligned} -\log \mathcal{L}\left(\mathcal{X}\prime, \mathcal{X}, \mathcal{C}, g_l; \mathcal{Z}, J, \mathcal{Q}\right) &= \frac{1}{2\sigma^2} \sum_{i=1}^{\boldsymbol{c}} \sum_{k\prime=1}^{K\prime_i} \left\| \mathsf{z}\prime_{ik\prime} - h\left(\boldsymbol{c}_i, \mathsf{x}\prime_{\mathbf{j}\prime_{ik\prime}}\right)\right\|^2 \\ &+ \frac{1}{2\sigma^2} \sum_{i=1}^{\boldsymbol{c}} \sum_{k=1}^{K_i} \left\| \mathsf{z}_{ik} - h_s\left(\boldsymbol{c}_i, \mathsf{x}_{\mathbf{j}_{ik}}\right)\right\|^2 \end{aligned} \tag{6.23}$$

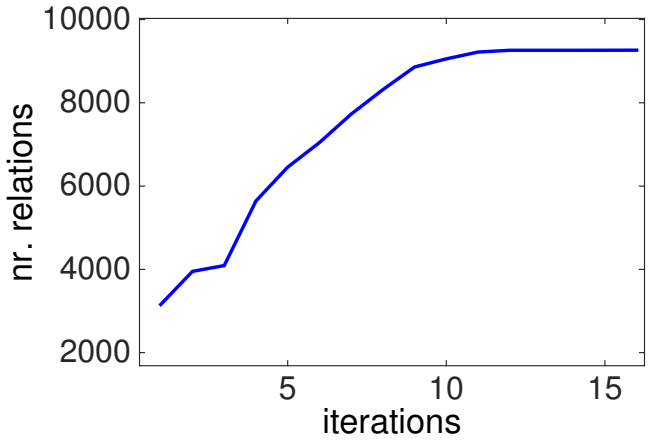
Guided Matching

From Equation 6.23, we can see that for every virtual point x_j generating a measurement z_{ik} in camera c_i we need the Miller indices of the corresponding point that is being virtually (or really) observed. This information is modeled using the variable $\mathbf{j}_{ik}$.

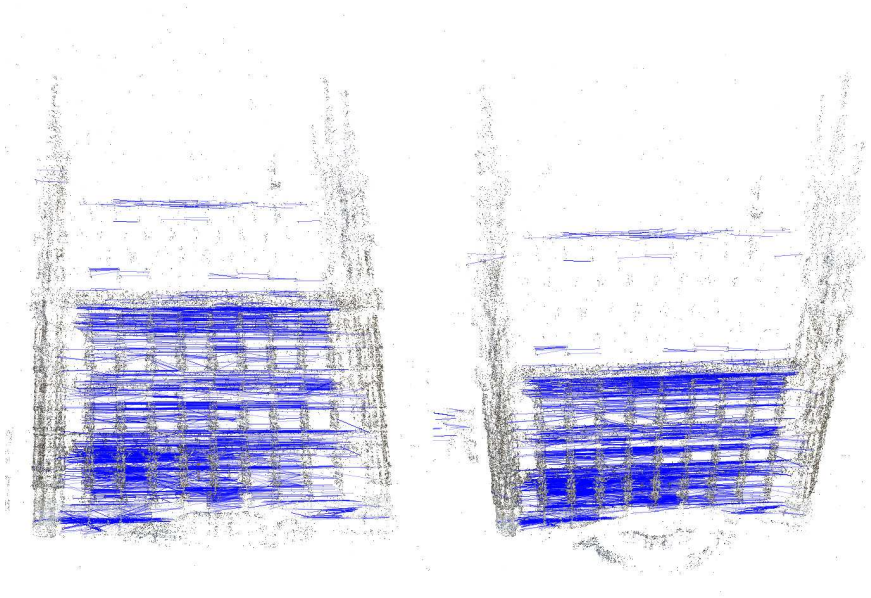
We get the correspondence information and the Miller indices using a guided matching scheme as follows. If point x_j is being observed was originally observed by camera c_i. We now transform the point as

$$\mathsf{x}_{\mathsf{q}_{j1}\mathsf{q}_2\ldots\mathsf{q}_{jL}} = \left({}_l\mathsf{g}_1^{\mathsf{q}_{j1}} \circ {}_l\mathsf{g}_2^{\mathsf{q}_{j2}} \circ \ldots {}_l\mathsf{g}_L^{\mathsf{q}_{jL}}\right) \mathsf{x}_j \; \forall \; -Q_{j1} \leq \mathsf{q}_{j1} \leq Q_{j1} \ldots$$

where $Q_{j1} \ldots Q_{jL}$ is predetermined to a reasonable value ~ 10 or the maximum number of elements if of the lattice that is being observed by the cameras. Every points $\mathsf{x}_{j\mathsf{q}_{j1}\mathsf{q}_{j2}\ldots\mathsf{q}_{jL}}$ is then projected to all the cameras (This also includes the original measurement which has a miller index of 0). We then look for a descriptor match between original feature in camera c_i and a small region of radius r that is centered around the projection



(a) Inliers vs Iteration



(b)

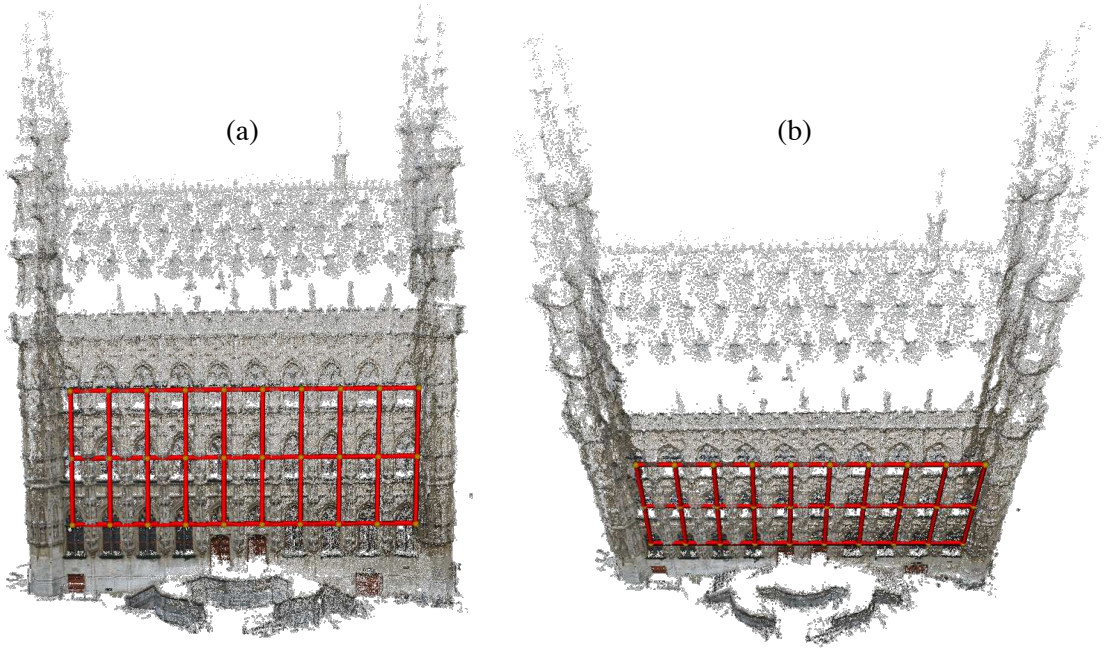
Figure 6.13: **Quantitative Evaluation: Iterative Guided Matching and Optimization:** I evaluate the quality of the reconstruction by counting the number of 3D inliers in the guided matching scheme. With every iteration, we get a better estimate of generators of lattice g_l which in turn leads to more matches between 3D points. (a) x-axis represents the iteration cycle of the whole guided matching and optimization scheme and on the y-axis, we have the number of inliers (matches) (b) Visualization of the matches on the final optimization (iteration = 15) or until there is no more increase in number of matches.

$\Pi\left({}_w\mathsf{R}_c\mathsf{x}_{\mathsf{q}_{j1}\mathsf{q}_{j2}\ldots\mathsf{q}_{jL}} - {}_w\,\mathsf{t}_c\right)$ for all cameras $c \in \mathcal{C}$. If we find a match, we update the measurements on camera c with a new z_{ik} that has a correspondence j_{ik} and a Miller Index $\mathsf{q}_{\mathsf{j}_{ik}} = [\mathsf{q}_{j1}\mathsf{q}_2 \ldots \mathsf{q}_{jL}]$.

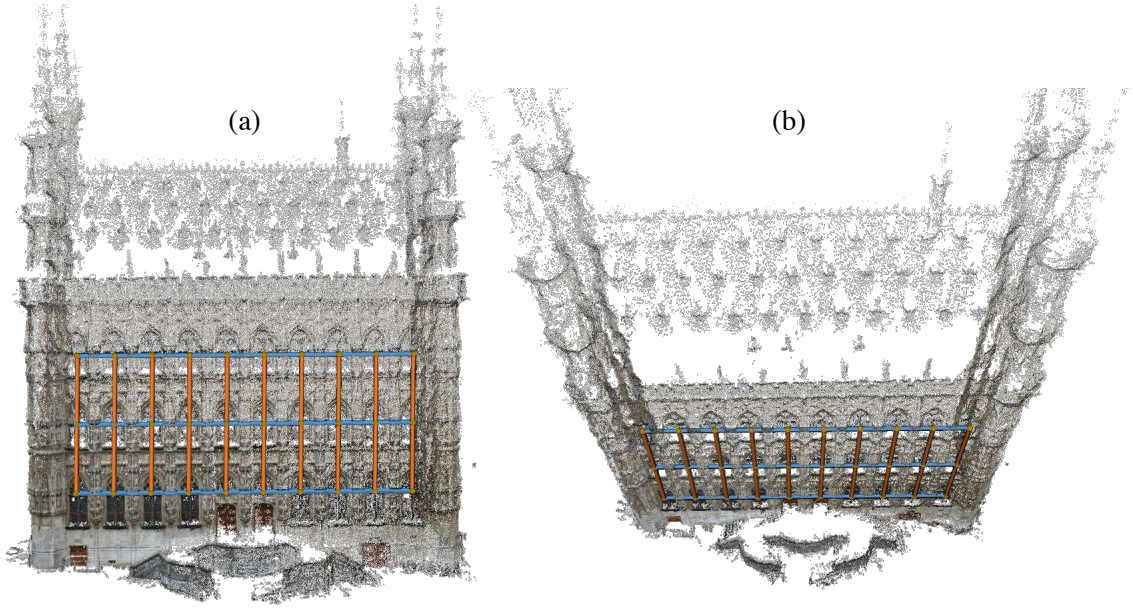
Results

Figure 6.14 and Figure 6.15 show the refined lattices for the Leuven and Neptune dataset. We have already defined the Leuven dataset in the previous section. I use a second dataset that I used in the previous chapter, the Neptune dataset which includes 70 pictures of the Temple of Neptune, in Paestum, Italy. The point cloud contains 48835 points.

We optimize the energy function given in Equation 6.23 using the Levenberg-Marquardt method, which outputs an improved estimate for the 3D points (and the cameras) and a refined estimate for the generators; analytic expressions of the involved Jacobians are given in Appendix A. Since the optimization refined the 3D model, it possibly exposed other measurements belonging to the lattice, hence we repeat the $n-$fold repetition discovery described earlier in this section, trying to expand the set of measurements z_{ik}. Indeed, Figure 6.13(a) shows that the number of symmetry relations increases monotonically with every iteration cycle of the guided matching-optimization, meaning that more and more points in the lattice are "discovered".

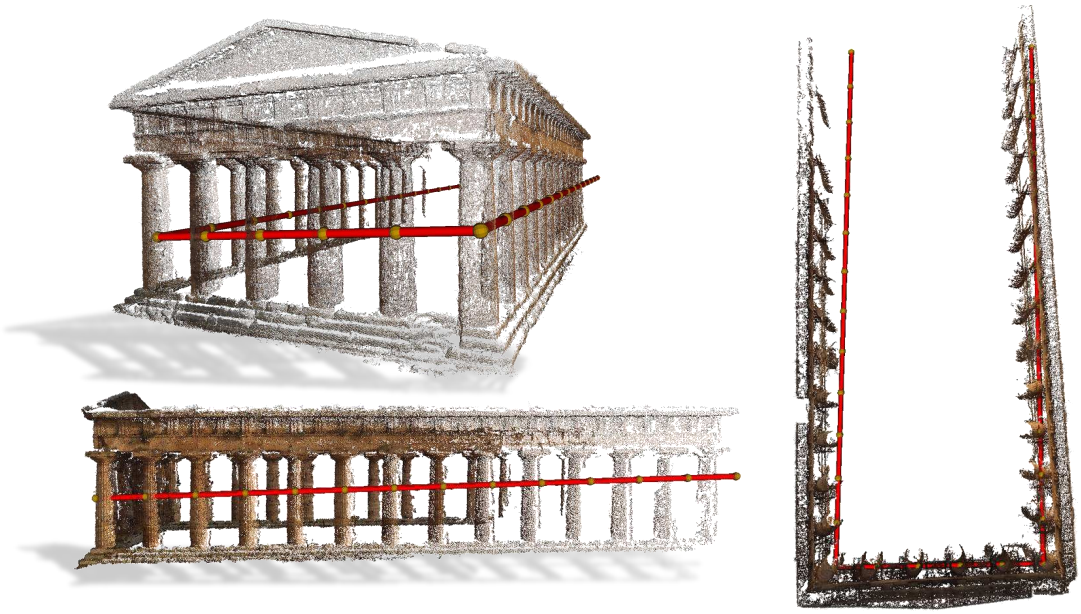


(a) [**Leuven Dataset**]: Estimated lattice using voting only

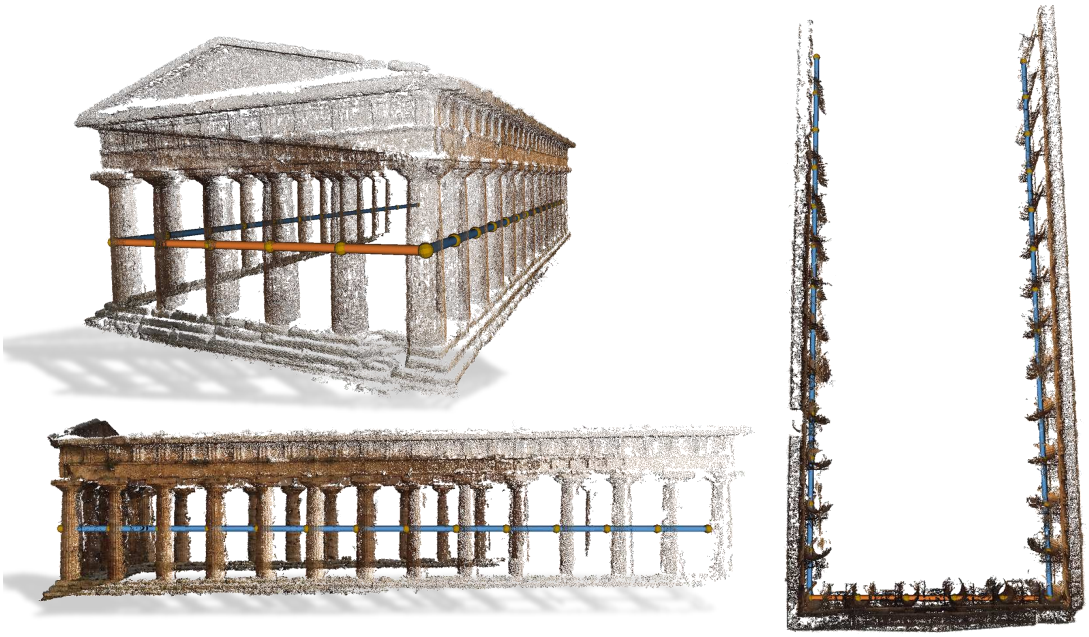


(b) [**Leuven Dataset**]: Refined lattice and structure.

Figure 6.14: **Refinement of Lattice using SfM** [**Leuven**]: [**Top Row**] Output of polar voting scheme on SfM point clouds to determine the generators of lattice $g_l = \{\mathrm{g}_1, \mathrm{g}_2\}$. The generators have errors in them, which becomes apparent when overlay-ed on the point clouds. (see right-most and left-most points of the lattice) Starting and ending locations do not coincide. Also the plane of the lattice is not fully parallel to the plane of the building. [**Bottom Row**] Lattice and 3D geometry optimized using technique outlined in Section 6.7.1.



(a) [**Neptune Dataset**]: Estimated lattice using voting only



(b) [**Neptune Dataset**]: Refined lattice and structure.

Figure 6.15: **Refinement of Lattice using SfM** [**Neptune**]: More results, the Neptune dataset better illustrates the benefit of lattice refinement along with SfM. [**Top Row**] Output of polar voting scheme on SfM point clouds to determine the generators of lattice $g_l = \{\mathrm{g}_1, \mathrm{g}_2\}$. The generators have errors in them, which becomes apparent when overlayed on the point clouds. (see right-most and left-most points of the lattice) Starting and ending locations do not coincide. Also the plane of the lattice is not fully parallel to the plane of the building. [**Bottom Row**] Lattice and 3D geometry optimized using technique outlined in Section 6.7.1.

Conclusion

I conclude this chapter by revisiting the claim that this chapter addresses, which is, to demonstrate that multi-view constraints can improve the reconstruction and symmetry parameters that are determined from a single view. Multi-view reconstruction under symmetry is still a largely unsolved problem and I have shown how one can solve this problem in two popular configurations (1) when we can do SfM first and (2) when we cannot do SfM first. I have shown much more denser asymmetric units and 3D models as a consequence of increased number of measurements from multiple views while simultaneously refining the location of the cameras.

I further show that we can atleast obtain a subset of the generative model of symmetry, specifically, the lattice parameters **without** any manual intervention. This is an extension of the voting scheme that we discussed in Chapter 4 which nicely brings together the different components of this thesis and allows the reader to chose the right algorithm for implementation depending on his/her constraints.

Chapter 7

CONCLUSION

In this thesis, we have seen the different challenges that we faced in modeling symmetries for urban reconstruction. The categories/types of symmetries, while apparently diverse, is still composed of a small set of well-defined fundamental operations. The mathematical theory of symmetry discusses the geometric and the group-theoretic aspect of this relationship and it affords us a unique opportunity to diversify the range of symmetries that we can analyze. I have explored this aspect of symmetry in this thesis. It has been enabled because of the positive results of the steps taken in this direction by Liu *et al.* [LCT04], who first demonstrated the benefit of thinking about symmetry from this perspective in the computer vision community for the case of 2D symmetries.

I first provided a framework for modeling complex symmetries in 1D, 2D and 3D using the mathematical theory of symmetry and lattices. This is especially important in the computer vision community where symmetry is often vaguely considered as some form of repetition, while in fact, it is much more than that. A result of rigorous modeling of symmetries paved the way to a classification scheme for symmetries of the 3D world. This categorization is especially useful to analyze the literature of symmetry in computer vision, because it tells

us which areas of symmetry analysis are lacking. A potential future work could be to use this classification scheme to segment the computer vision and computer graphics literature into different blocks. We provide a part of this classification in this thesis, but it can be far more extensive.

I then developed a probabilistic modeling framework in the form of a Bayes' Net that represents a generative model of symmetry. By doing so one can see the physical meaning of the variables that are generating the symmetry. The probabilistic nature of the model allows us to incorporate any prior knowledge of these variables using the interpretation of their physical meaning. For each case of classification, I show how we can use this probabilistic modeling framework for optimizing the parameters of symmetry using simulated observations in the form of points on the symmetric structure. The Bayes' net for modeling pixels as measurements is particularly useful to the computer vision community and the generative model that I provide directly serves this purpose.

In the third part of my thesis, I link the mathematical equivalence of the generators of discrete translational symmetry and the lattice parameters that we introduced in the previous chapter. By doing so, I was able to successfully exploit a novel voting scheme to determine the lattice parameters alone while decoupling the other elements of symmetry. This technique is particularly useful because it is completely automatic and works on even noisy reconstructed point clouds. This technique can also be really useful for the Computer Graphics community where voting as a means to determine symmetry in the 3D structure is quite popular.

In the last two section of my thesis, I discuss the interaction of the generative model of symmetry in an image projection space and show how we can infer the complex 3D sym-

metry of the scene as a collection of 3D points using only image pixels as measurements. By using the generative modeling introduced earlier, I was able to obtain a detailed 3D reconstruction of the scene while also determining the parameters of the camera that is imaging the scene, all from a single image. It was interesting to see the level of detail about the structure that can be directly determined from only a single image, a technique that is highly useful for a variety of tasks in computer vision where determining the depth of outdoor scenes is challenging even with multiple images. However, I would have liked to make this technique fully automatic as the semi-automatic nature of the proposed work somewhat diminishes its utility. Recent works in the area of symmetry detection show a positive trend in detecting complex 3D symmetries such as the methods compared here [Fun+17] thanks in part to the development of learning techniques.

In the last part of my thesis, I address the topic of joint SfM and symmetry detection. Structure from motion is currently an indispensable tool in computer vision. However, it is still not sufficiently agnostic to the type of environment that we are trying to reconstruct. Highly symmetric scenes are one such problem. In most cases, modern SfM algorithms are able to at least obtain a sparse reconstruction of the scene by discarding away a majority of matches in the symmetric regions as outliers in the geometric filtering stage. However, I show that symmetry can be a powerful constraint in SfM providing us with dense and photo-realistic 3D model of the scene. In cases where SfM completely fails, we show that we can still recover a plausible model of the scene that can be useful as a representation of the relative geometry of the scene. Similar to the case above, it would be useful to make this technique completely automatic by exploiting a symmetry-based recognition and segmentation method that can identify complex 3D symmetries from images.

I conclude my thesis by saying that symmetry can be a useful tool in modeling urban scenes using images. This has been repeatedly demonstrated by many authors, and I stand on the shoulders of these works to see a little farther by modeling complex symmetries in a methodical fashion. I believe that my work can be useful for the computer vision community to effectively identify new areas of research and to also develop new computer vision challenges such as those described here [Fun+17] while also providing a benchmarking scheme using the classification criteria that I provide.

Appendix A

APPENDIX A

Mathematical Notations

In this thesis, we use the following notations and conventions.

Rules

- Mathematical Spaces are denoted by **uppercase Blackboard** bold style.($\mathbb{R}$)
- Sets are represented denoted by **uppercase Calligraphic** style. ($\mathcal{R}$)
- Vectors are represented by the **lowercase Sans Serif** fonts. (r)
- Matrices are denoted by the **uppercase Sans Serif** font. (R)
- Indices represented by a variable r usually span from $1 \dots R$; the uppercase of the index.

Values

- A point $\mathsf{x} \in \mathbb{R}^n$ is a vector given by $\mathsf{x} = [x_1, x_2 \ldots \mathsf{x}_n]^T$
- A point in 2D $\mathsf{x} \in \mathbb{R}^2$ is an exception $\mathsf{x} = [x, y]^T$ to keep popular notation.
- A point in 3D $\mathsf{x} \in \mathbb{R}^3$ is an exception $\mathsf{x} = [x, y, z]^T$ to keep popular notation.
- A point in a $k-$dimensional lattice is $\mathsf{u''} \in \mathbb{L}^k$
- A point in a unit cell in a $k-$dimensional lattice $0 \leq \mathsf{u'} \leq 1$
- The set of points, accordance with the rules are represented by $\mathcal{X}$,$\mathcal{U''}$,$\mathcal{U'}$ for each of the above cases respectively.
- Measurements from a sensor are denoted by the vector z
- The set of all measurements is given by $\mathcal{Z}$
- The Miller index is a vector index of a point in a $k-$dimensional lattice represented by $\mathsf{q} \in \mathbb{Z}^k$
- If there is no bounds on the miller index, its range is the entire set of integers $\mathbb{Z}^k$
- If we choose to bound the index, we can represent $\mathsf{q} = \{-Q \ldots Q\}$ where Q is user defined based on the maximum number of allowable repetitions along each of the k axes. For example, if we allow repetitions of $\{Q_1, Q_2 \ldots Q_k\}$ along the k axes, then q is one of $(2Q_1 + 1)\,(2Q_2 + 1) \ldots (2Q_k + 1)$ entries where a particular entry $\mathsf{q} = [q_1, q_2 \ldots q_k]^T$ is such that $-Q_1 \leq q_1 \leq Q_1, -Q_2 \leq q_2 \leq Q_2 \ldots - Q_k \leq q_k \leq Q_k$

- If the Miller indices q are known for a set of points in a lattice $\mathcal{U}$ the set of corresponding miller indices is given by $\mathcal{Q}$

REFERENCES

[AD04] F. Alegre and F. Dellaert. "A Probabilistic Approach to the Semantic Interpretation of Building Facades". In: *International Workshop on Vision Techniques Applied to the Rehabilitation of City Centres*. International Committee for Architectural Photogrammetry; (CIPA); Lisbon. Oct. 2004.

[Aga+11] S. Agarwal et al. "Building Rome in a day". In: *Communications of the ACM* 54.10 (2011).

[Bao+13] Sid Bao et al. "Dense object reconstruction with semantic priors". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2013, pp. 1264–1271.

[BBI01] Dmitri Burago, Yuri Burago, and Sergei Ivanov. *A course in metric geometry*. Vol. 33. American Mathematical Society Providence, RI, 2001.

[BCK03] John Bamberg, Grant Cairns, and Devin Kilminster. "The crystallographic restriction, permutations, and Goldbach's conjecture". In: *The American mathematical monthly* 110.3 (2003), pp. 202–209.

[Ber+08] A. Berner et al. "A graph-based approach to symmetry detection". In: *In Proc. of Symp. on Volume and Point-Based Graphics*. 2008.

[BGM07] A.C. Berg, F. Grabler, and J. Malik. "Parsing images of architectural scenes". In: *Intl. Conf. on Computer Vision (ICCV)*. 2007, pp. 1–8.

[BH09] S. Becker and N. Haala. "Grammar supported facade reconstruction from mobile LIDAR mapping". In: *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Commission XXXVII, Paris, France* (2009).

[BR06] C. Brenner and N. Ripperda. “Extraction of facades using RJMCMC and constraint equations”. In: *Photogrammetric Computer Vision* (2006), pp. 155–160.

[BS11] Sid Yingze Bao and Silvio Savarese. “Semantic structure from motion”. In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE. 2011, pp. 2025–2032.

[CC94] T. Cham and R. Cipolla. “A local approach to recovering global skewed symmetry”. In: *IAPR*. Vol. 1. 1994, pp. 222–226.

[Cey+12] D. Ceylan et al. “Factored Facade Acquisition using symmetric line arrangements”. In: *Computer Graphics Forum (EUROGRAPHICS)*. Vol. 31. 2012, pp. 671–680.

[Cey+14] Duygu Ceylan et al. “Coupled structure-from-motion and 3D symmetry detection for urban facades”. In: *ACM Transactions on Graphics (TOG)* 33.1 (2014), p. 2.

[Cey14] Duygu Ceylan. “Computational Shape Understanding for 3D Reconstruction and Modeling”. PhD thesis. EPFL, 2014.

[CH80] R.W. Conners and C.A. Harlow. “Toward a Structural Textural Analyzer Based on Statistical Methods”. In: *Proc. Intl. Conf. Color in Graphics and Image Processing*. Vol. 12. 1980.

[CN89] Richard W Conners and Chong T Ng. “Developing a quantitative model of human preattentive vision”. In: *Systems, Man and Cybernetics, IEEE Transactions on* 19.6 (1989), pp. 1384–1407.

[Coh+12] Andrea Cohen et al. “Discovering and exploiting 3D symmetries in structure from motion”. In: Providence, RH, USA, June 2012.

[Coh+17] Andrea Cohen et al. “Symmetry-Aware Façade Parsing with Occlusions”. In: *3D Vision (3DV), 2017 International Conference on*. IEEE. 2017, pp. 393–401.

[CS13] John Horton Conway and Neil James Alexander Sloane. *Sphere packings, lattices and groups*. Vol. 290. Springer Science & Business Media, 2013.

[CT99] S. Coorg and S. Teller. “Extracting Textured Vertical Facades from Controlled Close-Range Imagery”. In: *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. 1999, pp. 625–632.

[Dai+12] D. Dai et al. “Learning Domain Knowledge for Facade Labelling”. In: *European Conf. on Computer Vision (ECCV)*. Firenze, Italy, Oct. 2012.

[Del+00] F. Dellaert et al. “Structure from Motion without Correspondence”. In: *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. June 2000.

[DK17] Frank Dellaert and Michael Kaess. “Factor Graphs for Robot Perception”. In: *Foundations and TrendsÂ® in Robotics* 6.1-2 (2017), pp. 1–139.

[DTM96] P.E. Debevec, C.J. Taylor, and J. Malik. “Modeling and Rendering Architecture from Photographs: A Hybrid Geometry- and Image-Based Approach”. In: *SIGGRAPH* 30 (1996), pp. 11–20.

[FP10] Y. Furukawa and J. Ponce. “Accurate, dense, and robust multi-view stereopsis”. In: *IEEE Trans. Pattern Anal. Machine Intell.* 32.8 (2010), pp. 1362–1376.

[FS11] Sam Friedman and Ioannis Stamos. “Real Time Detection of Repeated Structures in Point Clouds of Urban Scenes.” In: *3DIMPVT*. Ed. by Michael Goesele et al. IEEE, 2011, pp. 220–227.

[Fun+17] Christopher Funk et al. “2017 ICCV Challenge: Detecting Symmetry in the Wild”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 1692–1701.

[GB94] A.D. Gross and T.E. Boult. “Analyzing Skewed Symmetries”. In: *IJCV* 13.1 (1994).

[GC06] R. Gal and D. Cohen-Or. “Salient geometric features for partial shape matching and similarity”. In: *ACM Trans. on Graphics* 25.1 (2006).

[GMP96] L. Van Gool, T. Moons, and M. Proesmans. “Mirror and Point Symmetry under Perspective Skewing”. In: *CVPR*. 1996.

[Hay+06] J.H. Hays et al. “Discovering Texture Regularity as a Higher-Order Correspondence Problem”. In: *European Conf. on Computer Vision (ECCV)*. 2006.

[Hon+04] Wei Hong et al. “On symmetry and multiple-view geometry: Structure, pose, and calibration from a single image”. In: *International Journal of Computer Vision* 60.3 (2004), pp. 241–265.

[HS12] Daniel Cabrini Hauagge and Noah Snavely. “Image matching using local symmetry features”. In: *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE. 2012, pp. 206–213.

[HZ00] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[HZ03] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[Jia+15] Nianjuan Jiang et al. “Direct Structure Estimation for 3D Reconstruction”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 2655–2663.

[JTC09] Nianjuan Jiang, Ping Tan, and Loong-Fah Cheong. “Symmetric architecture modeling with a single image”. In: *ACM Transactions on Graphics (TOG)* 28.5 (2009), p. 113.

[JTC11] N. Jiang, P. Tan, and L.-F. Cheong. “Multi-view repetitive structure detection”. In: *ICCV*. 2011, pp. 535–542.

[Kan81] T. Kanade. “Recovery of the 3-Dimensional Shape of an Object from a Single View”. In: *Artificial Intelligence* 17 (1981).

[Kas+00] Margaret E Kastner et al. *Crystallographic CourseWare*. 2000.

[Kaz+03] M. Kazhdan et al. “A reflective symmetry descriptor for 3d models”. In: *Algorithmica* 38 (2003).

[Kha12] Ilyas Khan. “HYPERBOLIC GEOMETRY: ISOMETRY GROUPS OF HYPERBOLIC SPACE”. In: (2012).

[KK07] Kevin KÃ¶ser and Reinhard Koch. “Perspectively invariant normal features”. In: *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE. 2007, pp. 1–8.

[KK97] T. Kanade and J.R. Kender. “Mapping Image Properties into Shape Constraints: Skewed Symmetry, Affine-Transformable Patterns, and the Shape-from-Texture Paradigm”. In: *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms, M.A. Fischler and O. Firschein, eds.* (1997).

[Koh+16] Peter J Kohler et al. “Representation of maximally regular textures in human visual cortex”. In: *Journal of Neuroscience* 36.3 (2016), pp. 714–729.

[KZP11] Kevin Köser, Christopher Zach, and Marc Pollefeys. “Dense 3d reconstruction of symmetric scenes from a single image”. In: *Joint Pattern Recognition Symposium*. Springer. 2011, pp. 266–275.

[LCL08] S. Lee, R. Collins, and Y. Liu. “Rotation Symmetry Group Detection Via Frequency Analysis of Frieze-Expansions”. In: *CVPR*. 2008.

[LCT04] Y. Liu, R. Collins, and Y. Tsin. “A Computational Model for Periodic Pattern Perception Based on Frieze and Wallpaper Groups”. In: *IEEE Trans. Pattern Anal. Machine Intell.* 26.3 (Mar. 2004), pp. 354–371.

[LE06] Gareth Loy and Jan-Olof Eklundh. “Detecting symmetry and symmetric constellations of features”. In: *Computer Vision–ECCV 2006*. Springer, 2006, pp. 508–521.

[Lip+10] Y. Lipman et al. “Symmetry factored embedding and distance”. In: *SIGGRAPH*. 2010.

[Liu+05] Y. Liu et al. “Digital papercutting”. In: *SIGGRAPH*. 2005.

[Liu+10] Y. Liu et al. “Computational Symmetry in Computer Vision and Computer Graphics”. In: *Foundations and Trends in Computer Graphics and Vision* 5.1–2 (2010).

[LL07] Wen-Chieh Lin and Yanxi Liu. “A lattice-based MRF model for dynamic near-regular texture tracking”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29.5 (2007), pp. 777–792.

[LL10] Seungkyu Lee and Yanxi Liu. “Skewed rotation symmetry group detection”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.9 (2010), pp. 1659–1672.

[LL13] Jingchen Liu and Yanxi Liu. “Grasp recurring patterns from a single view”. In: *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE. 2013, pp. 2003–2010.

[LLH04] Yanxi Liu, Wen-Chieh Lin, and James Hays. “Near-regular texture analysis and manipulation”. In: *ACM Transactions on Graphics (TOG)* 23.3 (2004), pp. 368–376.

[LM96] T. Leung and J. Malik. “Detecting, Localizing and Grouping Repeated Scene Elements”. In: *ECCV*. Vol. 1. 1996.

[Low99] David G Lowe. “Object recognition from local scale-invariant features”. In: *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*. Vol. 2. IEEE. 1999, pp. 1150–1157.

[LSD13] Alex Levinshtein, Cristian Sminchisescu, and Sven Dickinson. “Multiscale symmetric part detection and grouping”. In: *International journal of computer vision* 104.2 (2013), pp. 117–134.

[LWY97] H.-C. Lin, L.-L. Wang, and S.-N. Yang. “Extracting Periodicity of a Regular Texture Based on Autocorrelation Functions”. In: *Pattern Recognition Letters* 18 (1997).

[Mar+06] A. Martinet et al. “Accurate detection of symmetries in 3d shapes”. In: *ACM Trans. on Graphics* 25.2 (2006).

[Mar12] George Edward Martin. *The foundations of geometry and the non-Euclidean plane*. Springer Science & Business Media, 2012.

[McW02] Roy McWeeny. *Symmetry: An introduction to group theory and its applications*. Courier Corporation, 2002.

[MFR04] M. Mazhdan, T. Funkhouser, and S. Rusinkiewicz. "Symmetry descriptors and 3D shape matching". In: *In Proc. of Symp. of Geometry Processing*. 2004.

[MGP06] Niloy J. Mitra, Leonidas J. Guibas, and Mark Pauly. "Partial and Approximate Symmetry Detection for 3D Geometry". In: *ACM SIGGRAPH 2006 Papers*. SIGGRAPH '06. Boston, Massachusetts: ACM, 2006, pp. 560–568. ISBN: 1-59593-364-6.

[Mit+12] N.J. Mitra et al. "Symmetry in 3D Geometry: Extraction and Applications". In: *EUROGRAPHICS State-of-the-art Report*. 2012.

[Mül+06] Pascal Müller et al. "Procedural Modeling of Buildings". In: *ACM Trans. Graph.* 25.3 (July 2006), pp. 614–623.

[Mül13] Ulrich Müller. *Symmetry relationships between crystal structures: applications of crystallographic group theory in crystal chemistry*. Vol. 18. OUP Oxford, 2013.

[MZB95] D. Mukherjee, A. Zisserman, and J. Brady. "Shape from symmetry—detecting and exploiting symmetry in affine images". In: *Phil. Trans. R. Soc. Lond. A* 351 (1995), pp. 77–106.

[Nia12] Jiang Nianjuan. "A study of Symmetric and Repetitive Structures in Image-Based Modeling". PhD thesis. 2012.

[NLD11] Richard A Newcombe, Steven J Lovegrove, and Andrew J Davison. "DTAM: Dense tracking and mapping in real-time". In: *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE. 2011, pp. 2320–2327.

[Par+09] Minwoo Park et al. "Deformed lattice detection in real-world images using mean-shift belief propagation". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 31.10 (2009), pp. 1804–1816.

[Pau+08a] Mark Pauly et al. "Discovering structural regularity in 3D geometry". In: *ACM Transactions on Graphics (TOG)*. Vol. 27. 3. ACM. 2008, p. 43.

[Pau+08b] M. Pauly et al. “Discovering Structural Regularity in 3D Geometry”. In: *ACM Transactions on Graphics* 27.3 (2008), #43, 1–11.

[PD05] V.S.N. Prasad and L.S. Davis. “Detecting rotational symmetries”. In: *ICCV*. Vol. 2. 2005, pp. 954–961.

[Pea88] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[PGR07] J. Podolak, A. Golovinskiy, and S. Rusinkiewicz. “Symmetry-enhanced remeshing of surfaces.” In: *In Proc. of Symp. of Geometry Processing*. 2007.

[PL94] S.C. Pei and L.G. Liou. “Automatic Symmetry Determination and Normalization for Rotationally Symmetric 2D Shapes and 3D Solid Objects”. In: *Pattern Recognition* 27 (1994).

[Rav+07] D. Raviv et al. “Symmetries of non-rigid shapes”. In: *ICCV*. 2007.

[Rob+11] Richard Roberts et al. “Structure from motion for scenes with large duplicate structures”. In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE. 2011, pp. 3137–3144.

[Rot+06] Fred Rothganger et al. “3d object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints”. In: *International Journal of Computer Vision* 66.3 (2006), pp. 231–259.

[Sal+14] Renato F Salas-Moreno et al. “Dense planar SLAM”. In: *Mixed and Augmented Reality (ISMAR), 2014 IEEE International Symposium on*. IEEE. 2014, pp. 157–164.

[Sas+05] Yuka Sasaki et al. “Symmetry activates extrastriate visual cortex in human and nonhuman primates”. In: *Proceedings of the National Academy of Sciences of the United States of America* 102.8 (2005), pp. 3159–3163.

[SCD15] Natesh Srinivasan, Luca Carlone, and Frank Dellaert. “Structural Symmetries from Motion for Scene Reconstruction and Understanding”. In: *Proceedings of the British Machine Vision Conference (BMVC)*. Ed. by Mark W. Jones

Xianghua Xie and Gary K. L. Tam. BMVA Press, Sept. 2015, pp. 136.1–136.13. ISBN: 1-901725-53-7.

[Sch+08] G. Schindler et al. “Detecting and Matching Repeated Patterns for Automatic Geo-tagging in Urban Environments”. In: *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2008.

[SD06] G. Schindler and F. Dellaert. “Line-Based Structure From Motion for Urban Environments”. In: *3DPVT*. 2006.

[Sha+14] Qi Shan et al. “Occluding contours for multi-view stereo”. In: *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE. 2014, pp. 4002–4009.

[Sna+10] Noah Snavely et al. “Bundler: Structure from motion (SFM) for unordered image collections”. In: *Available online: phototour. cs. washington. edu/bundler/(accessed on 12 July 2013)* (2010).

[SS97] C. Sun and J. Sherrah. “3d symmetry detection using the extended gaussian image”. In: *PAMI* 19.2 (1997).

[SSS09] Sudipta N Sinha, Drew Steedly, and Richard Szeliski. “Piecewise planar stereo for image-based rendering.” In: *ICCV*. 2009, pp. 1881–1888.

[Sun97] C.M. Sun. “Fast Recovery of Rotational Symmetry Parameters Using Gradient Orientation”. In: *Optical Eng.* 46.4 (1997).

[SX14] Shuran Song and Jianxiong Xiao. “Sliding shapes for 3D object detection in depth images”. In: *Computer Vision–ECCV 2014*. Springer, 2014, pp. 634–651.

[SZ99] F. Schaffalitzky and A. Zisserman. “Geometric Grouping of Repeated Elements within Images”. In: *Shape, Contour, and Grouping in Computer Vision, D.A. Forsyth, V. Di Gesu, J.L. Mundy, and R. Cipolla, eds., Springer-Verlag* (1999).

[TFA15] Ching L Teo, Cornelia Fermuller, and Yiannis Aloimonos. “Detection and segmentation of 2d curved reflection symmetric structures”. In: *Proceedings*

of the IEEE International Conference on Computer Vision. 2015, pp. 1644–1652.

[TK12] Stavros Tsogkas and Iasonas Kokkinos. "Learning-based symmetry detection in natural images". In: *European Conference on Computer Vision*. Springer. 2012, pp. 41–54.

[TMS09] G. Tzimiropoulos, N. Mitianoudis, and T. Stathaki. "A unifying approach to moment-based shape orientation and symmetry classification". In: *Trans. Img. Proc.* 18.1 (2009).

[TTG03] T. Tuytelaars, A. Turina, and L.V. Gool. "Non-combinatorial detection of regular repetitions under perspective skew". In: *PAMI* 25.4 (2003).

[Wat12] James Watson. *The double helix*. Hachette UK, 2012.

[WCL16] Mark Wolff, Robert T Collins, and Yanxi Liu. "Regularity-driven facade matching between aerial and street views". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 1591–1600.

[Wey15] Hermann Weyl. *Symmetry*. Princeton University Press, 2015.

[WFP10] Changchang Wu, Jan-Michael Frahm, and Marc Pollefeys. "Detecting Large Repetitive Structures with Salient Boundaries." In: *ECCV (2)*. Ed. by Kostas Daniilidis, Petros Maragos, and Nikos Paragios. Vol. 6312. Lecture Notes in Computer Science. Springer, 2010, pp. 142–155. ISBN: 978-3-642-15551-2.

[Wu+14] Fuzhang Wu et al. "Inverse Procedural Modeling of Facade Layouts". In: *ACM Trans. Graph.* 33.4 (July 2014), 121:1–121:10.

[Wu10] Changchang Wu. "Geometry-driven feature detection". PhD thesis. 2010.

[Yip+94] R.K.K. Yip et al. "Hough Transform Technique for the Detection of Rotational Symmetry". In: *Pattern Recognition Letters* 15.9 (1994).